Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1986

Aspects of simulating interstitial diffusion in a face-centered-cubic lattice.

Polnaszek, Mark R.

http://hdl.handle.net/10945/21685

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943-5002

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

ASPECTS OF SIMULATING INTERSTITIAL DIFFUSION IN A FACE-CENTERED-CUBIC LATTICE

by

Mark R. Polnaszek

June 1986

Thesis Advisor: Don E. Harrison, Jr.

Approved for public release; distribution is unlimited.

T232217

URITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
SECURITY CLASSIFICATION AUTHORITY DECLASSIFICATION / DOWNGRADING SCHEDULE	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
ERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)

NAME OF PERFORMING ORGANIZATION val Postgraduate Schhol	6b OFFICE SYMBOL (If applicable) 61	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
DDRESS (City, State, and ZIP Code) nterey, California 93943-5000		7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
NAME OF FUNDING / SPONSORING RGANIZATION	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

DDRESS (City, State, and ZIP Code)

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

ITLE (Include Security Classification) ASPECTS OF SIMULATING INTERSTITIAL DIFFUSION IN A FACE-CENTERED-CUBIC LATTICE.

ERSONAL AUTHOR(S) Polnaszek, Mark R.

TYPE OF REPORT sters Thesis	13b TIME COVERED FROM ________ TO ______	14 DATE OF REPORT (Year, Month, Day) June 1986	15 PAGE COUNT 101

UPPLEMENTARY NOTATION

COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
IELD	GROUP	SUB-GROUP	diffusion, simulation, Fick's Laws

BSTRACT (Continue on reverse if necessary and identify by block number) A simulation of interstitial diffusion hough a face-centered-cubic crystal was developed to investigate the positional eendence of the diffusion coefficient. The simulation is an improvement over similar yes of simulations because it keeps track of the nearest-neighbors and planar locations r arrays. These arrays enabled the simulation to investigate a more sophisticated system f diffusion, the face-centered-cubic crystal. The simulation provides output that can be sd to develop the diffusion coefficient by both Fick's first law method and Fick's eond law method. The simulation also allows one to determine a functional form for the ifusion coefficient if the positional dependence is indicated.

STRIBUTION / AVAILABILITY OF ABSTRACT ☒ NCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified	
AME OF RESPONSIBLE INDIVIDUAL Dn E. Harrison Jr.	22b TELEPHONE (Include Area Code) (408) 646-2877	22c OFFICE SYMBOL 61Hx

ORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

Approved for public release; distribution is unlimited.

Aspects of Simulating Interstitial Diffusion in a Face-Centered-Cubic Lattice.

by

Mark Ronald Polnaszek
Lieutenant, United States Navy
B.S., The Citadel, 1979

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN PHYSICS

from the

NAVAL POSTGRADUATE SCHOOL
June 1986

ABSTRACT

A simulation of interstitial diffusion through a face-centered-cubic crystal was developed to investigate the positional dependence of the diffusion coefficient. The simulation is an improvement over similar types of simulations because it keeps track of the nearest-neighbors and planar locations in arrays. These arrays enabled the simulation to investigate a more sophisticated system of diffusion, the face-centered-cubic crystal. The simulation provides output that can be used to develop the diffusion coefficient by both Fick's first law method and Fick's second law method. The simulation also allows one to determine a functional form for the diffusion coefficient if the positional dependence is indicated.

Thesis
P752
c.1

TABLE OF CONTENTS

I. INTRODUCTION -- 7

A. HISTORY -- 7

B. DIFFUSION MECHANISMS ---------------------------- 10

C. BASIC CONCEPTS ---------------------------------- 13

D. RANDOM WALKS ------------------------------------ 19

E. COMPUTER SIMULATION OF DIFFUSION ---------------- 19

F. CONCENTRATION DEPENDENT DIFFUSION COEFFICIENT 21

II. OBJECTIVE -- 23

III. COMPUTER MODEL AND SIMULATION -------------------- 26

A. PHYSICAL MODEL ---------------------------------- 27

B. COMPUTER SIMULATION ----------------------------- 29

C. RANDOM WALK ------------------------------------- 30

D. COMPUTER CODE ----------------------------------- 31

1. Difset --- 31

a. Crystal Set Up -------------------------- 31

b. Nearest-Neighbor Table ------------------ 34

c. Site vs Plane Table --------------------- 35

2. Diffuse -- 35

a. Concentration -------------------------- 37

b. Flux ------------------------------------ 37

c. First Derivative ----------------------- 38

d. Second Derivative ---------------------- 38

e. Time Derivative ------------------------ 39

f. Averaging ---------------------------- 39

g. Diffusion Coefficient ---------------- 40

IV. RESULTS --- 41

A. PROOF OF SIMULATION'S PHYSICAL EQUIVALENCE --- 41

1. Conservation of Matter ------------------- 41

2. Comparison of the Diffusion Coefficient -- 41

3. Concentration Profile --------------------- 43

B. CALCULATION OF THE DIFFUSION COEFFICIENT ----- 47

1. Fick's First Law Method ------------------ 47

2. Fick's Second Law Method ----------------- 51

3. Fokker-Plank Method ----------------------- 54

V. CONCLUSIONS AND RECOMMENDATIONS ------------------ 56

APPENDIX A: VARIABLE GLOSSARY ------------------------ 59

APPENDIX B: 'DIFSET' CODE LISTING -------------------- 63

APPENDIX C: 'DIFFUSE' CODE LISTING ------------------- 72

APPENDIX D: SAMPLE INPUT FOR 'DIFSET' ---------------- 79

APPENDIX E: SAMPLE INPUT FOR 'DIFFUSE' --------------- 81

APPENDIX F: SAMPLE OUTPUT OF 'DIFSET' ---------------- 86

APPENDIX G: SAMPLE OUTPUT OF 'DIFFUSE' --------------- 91

LIST OF REFERENCES ----------------------------------- 97

INITIAL DISTRIBUTION LIST ---------------------------- 100

LIST OF TABLES

1. PARTIAL TABLE OF DIFFUSION MECHANISMS ------------ 10
2. TABLE OF CRYSTAL STRUCTURES ---------------------- 14
3. TABLE OF DIFSET ARRAYS ---------------------------- 31
4. EXAMPLES OF ARRAY ENTRIES TO THE CRYST3 ARRAY ---- 32
5. EXAMPLE OF THE FIRST TWO PLANES IN A 4X4XN CRYSTAL 33
6. TABLE OF SHORT TIME-AVERAGED DIFFUSION COEFFICIENTS 52
7. TABLE OF LONG TIME-AVERAGED DIFFUSION COEFFICIENTS 52
8. TABLE OF SHORT TIME-AVERAGED TIME DERIVATIVES ---- 53
9. TABLE OF LONG TIME-AVERAGED TIME DERIVATIVES ----- 54

I. INTRODUCTION

A. HISTORY

Since man first poured a colored liquid into water and observed that the liquid spread throughout the container, since he placed two metals together in heat and later observed that they melded together, or since he first noticed that the smoke from his fire mixed into the air, diffusion has been observed and its cause a source of wonder.

Initial qualitative experiments in diffusion were conducted by Parrot in 1815 (Crank, 1975). In this experiment, he noted that no matter how carefully he controlled mechanical agitation and convection, the separate gases in the experiment always diffused throughout the container. One other important precursor development was Fourier's development of the equation for heat conduction (Fourier, 1822).

The history of the scientific study of diffusion is rooted in a paper by Adolf Fick (1855). Fick used Fourier's mathematical methods to describe the equations of diffusion. One equation states that the rate of diffusion is proportional to the concentration gradient across the plane of diffusion:

$$J = - D(\partial C / \partial x), \tag{1}$$

where J is the flux of diffusing material, D the diffusion coefficient, and C the concentration of the diffusing material. The units of the diffusion coefficient are (length)^2/(time), and it is usually given in centimeters^2 per second. Equation (1) is known as Fick's first law of diffusion. In this work, Equation (1), is assumed to be of this form, even for non-constant D.

Fick also developed a second equation which describes the time rate of change of the concentration of the diffusing material. Known as Fick's second law, it states that the partial derivative of the concentration with respect to time is equal to the negative of the divergence of the flux:

$$\partial C/\partial t = -\nabla J. \qquad (2)$$

In one dimensional cartesion coordinates, this reduces to:

$$\partial C/\partial t = -\ \partial J/\partial x. \qquad (3)$$

When Equation (3) and Fick's first law (Equation 1) are combined Equation (3) becomes:

$$\partial C/\partial t = \{\partial/\partial x[(D)(\partial C/\partial x)]\}, \qquad (3a)$$

and, if one assumes that the diffusion coefficient is constant with respect to position, the equation further reduces to:

$$\partial C/\partial t = D(\partial^2 C/\partial x^2). \qquad (3b)$$

The general solution to this equation, given an initial impulsive planar source which then spreads across a one-dimensional infinite material, is:

$$C = (A/t^{1/2})\exp(-x^2/4Dt), \tag{4}$$

where A is a constant, t the elapsed time, and x the position of interest within the material. A consequence of this solution is that after sufficient time the concentration will become uniform throughout the material as seen in Figure 1.

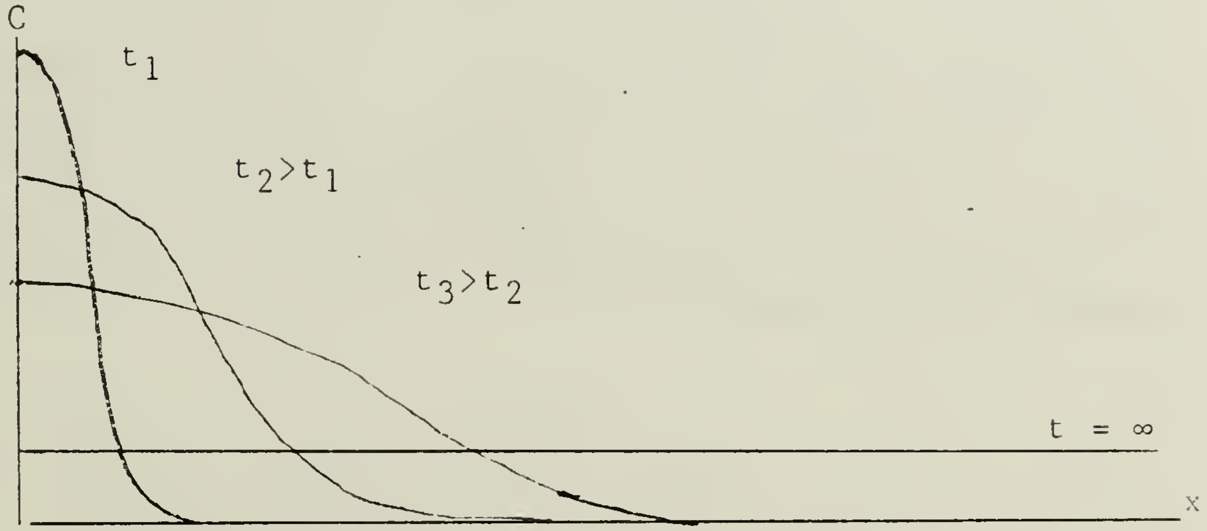


Figure 1. Concentration Profiles at Different Times

Einstein (1905) provided an atomistic definition of the diffusion coefficient:

$$D = \langle \underline{x}^2 \rangle / 2t, \tag{5}$$

where $\underline{x}$ is the vector displacement of a given atom from its initial position to its final position at time t. The Dirac brackets denote an average over a large number of diffusing atoms. It can readily be shown that the definitions for D in Equations (1) and (5) are equivalent.

B. DIFFUSION MECHANISMS

Since Fick's and Einstein's work in diffusion theory, many people have tried to explain more fully the mechanisms by which an atom diffuses through another material. The majority of this work was done by Kirkendall (1942), Hartly (1946) and Darken (1948). These men developed tools to explain diffusion based on a knowledge of the nature of the atom and the method by which atoms combine into bulk materials. Given the concept of the crystal structure of solids, many mechanisms of diffusion can be identified. Table 1 is a partial list of the mechanisms of diffusion through solid crystals:

TABLE 1

PARTIAL TABLE OF DIFFUSION MECHANISMS

1. Interstitial mechanism
2. Interstitialcy mechanism
3. Crowdion mechanism
4. Vacancy mechanism
5. Relaxion mechanism.

The INTERSTITIAL MECHANISM describes the diffusion of small atoms through a crystal structure of larger atoms. The atom advances through the structure by moving from one vacant intertitial site within the crystal to another. This is shown in Figure 2.

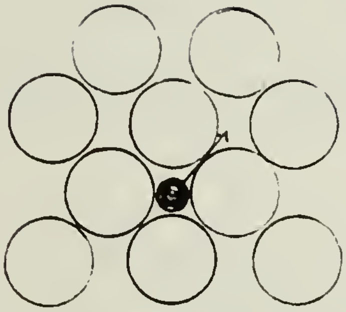


Figure 2. Intertitial Mechanism of Diffusion.

When the interstitial atom's size approaches that of the lattice atoms, they move by the INTERSTITIALCY MECHANISM. In this method the defect atom moves into a normal lattice position by pushing the lattice atom out into an interstitial site. Figure 3 illustrates this mechanism.

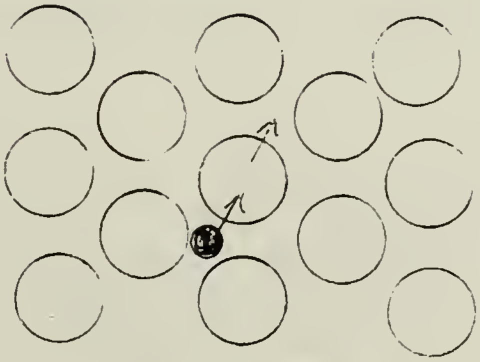


Figure 3. Intertitialcy Mechanism of Diffusion.

The CROWDION MECHANISM of diffusion requires a defect atom with an energy large enough such that it can push its way onto a lattice site,compressing a close packed row of atoms. This mechanism can be thought of as a line of billiard balls touching each other. Then a cue ball hits this line so that each ball moves one position, replacing its neighbor. An example is shown in Figure 4.

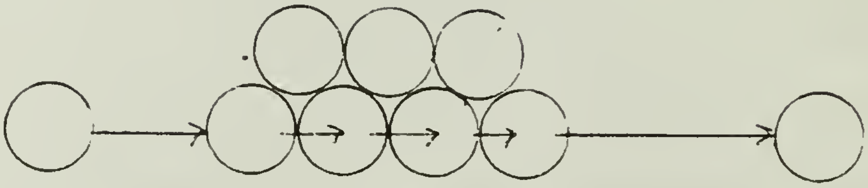


Figure 4. Crowdion Mechanism of Diffusion.

All crystals have vacancies in their lattices at a finite temperature. In the VACANCY MECHANISM of diffusion, these vacancies are filled by the diffusing atoms. This method of diffusion is predominant in self diffusion of a material since a lattice atom and the vacancy simply exchange position. Figure 5 illustrates the method by which the vacancy moves.

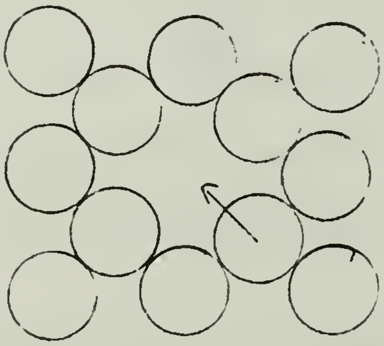


Figure 5. Vacancy Mechanism of Diffusion.

The RELAXION MECHANISM of diffusion is a modified vacancy mechanism. In the area of a vacancy the lattice atoms relax towards the vacancy. These atoms, and any defect atoms in this area, are able to move by an irregular movement similar to motion of a liquid. This mechanism can be seen in Figure 6.

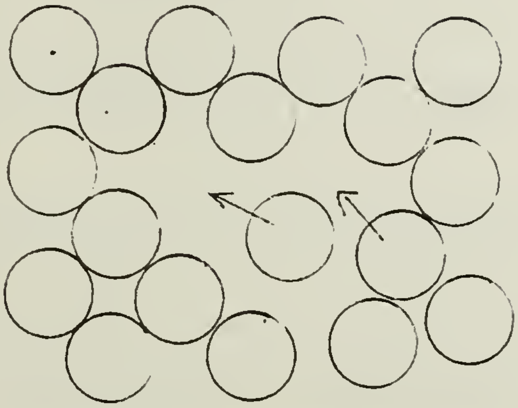


Figure 6. Relaxion Mechanism of Diffusion.

J.R. Manning (1968) has an excellent discussion of the mechanisms of diffusion.

C. BASIC CONCEPTS

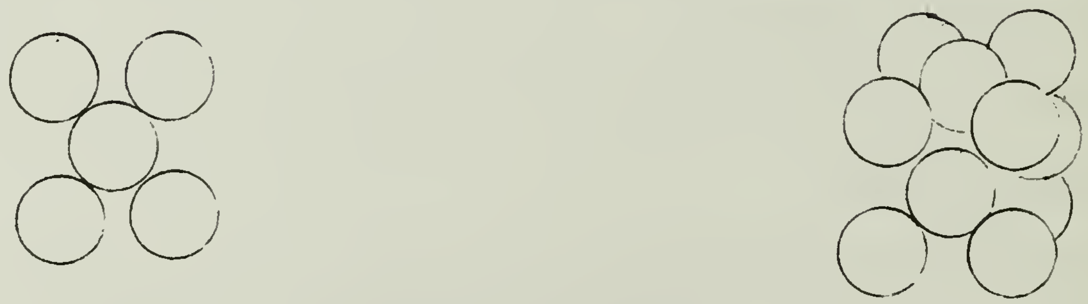
Crystaline solids are made up of atoms that arrange themselves in various lattice structures. There are seven types of crystals, these are:

TABLE 2

TABLE OF CRYSTAL STRUCTURE

1. cubic
2. tetragonal
3. orthorhombic
4. hexagonal
5. rhombohedral
6. monoclinic
7. triclinic

Within each of these types of crystals subdivisions exist that further divide these seven structures into a total of fourteen crystal structures. L.A. Girifalco (1964) gives a good discussion of the different crystal structures. The face-centered cubic (fcc) crystal structure is used in this thesis.

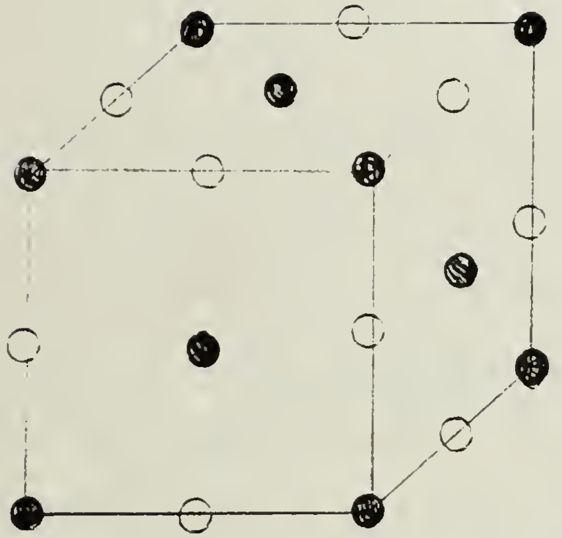


(1 0 0)

(3-D)

Figure 7. Different Propectives of the fcc Crystal.

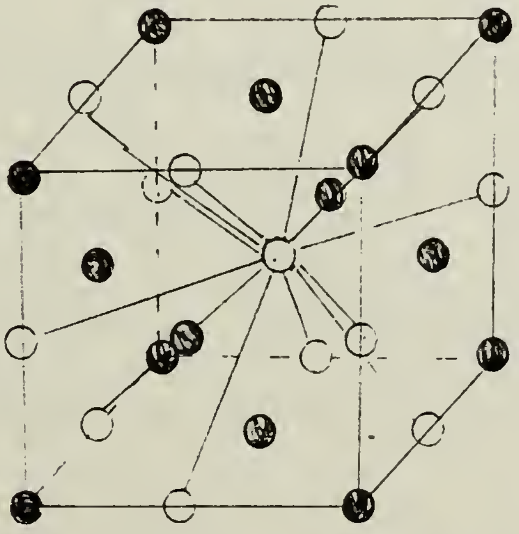
Figure 7 illustrates different sections of a fcc crystal. There exists sites between the atoms of the structure known as interstitial sites, see Figure 8.



○ -interstitial sites ● -lattice atoms

Figure 8. Intertitial and Lattice Sites of fcc Crystal.

Each site in the fcc structure, whether an interstitial site or lattice site, has twelve nearest neighbors, Figure 9 shows the relationship of a site and its nearest neighbors within the crystal. The interstitial method of diffusion described above moves an atom to one of these nearest-neighbor sites.



○ - interstitial sites ● - lattice sites

Figure 9. Fcc Atom and its Twelve Nearest Neighbors.

The force acting between a pair of atoms changes as the distance between the atoms changes. Initially, at large separations, the force is attractive, becoming stronger as they approach one another. At small separations the force become repulsive. This force is best described by a potential energy function. The minimum of this potential function is the equilibrium position between the pair of atoms. Figure 10 shows a representative plot of this potential energy for a pair of atoms.

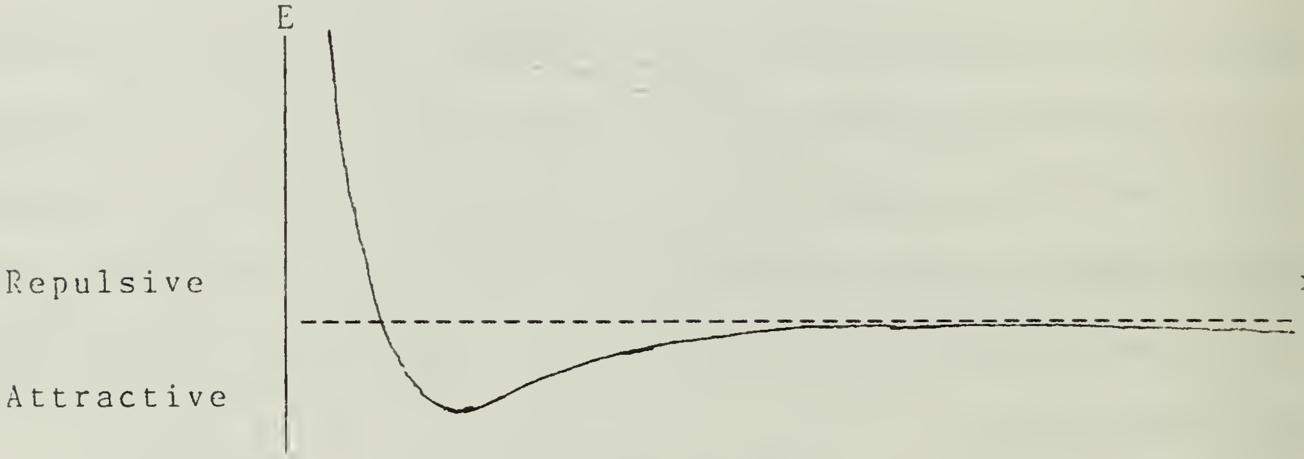


Figure 10. Representative Potential Energy Plot.

In a crystal structure this potential energy function is a multi-body problem. The result of these extra bodies is that at a greater distance, the function oscillates about the zero energy position as shown in Figure 11.

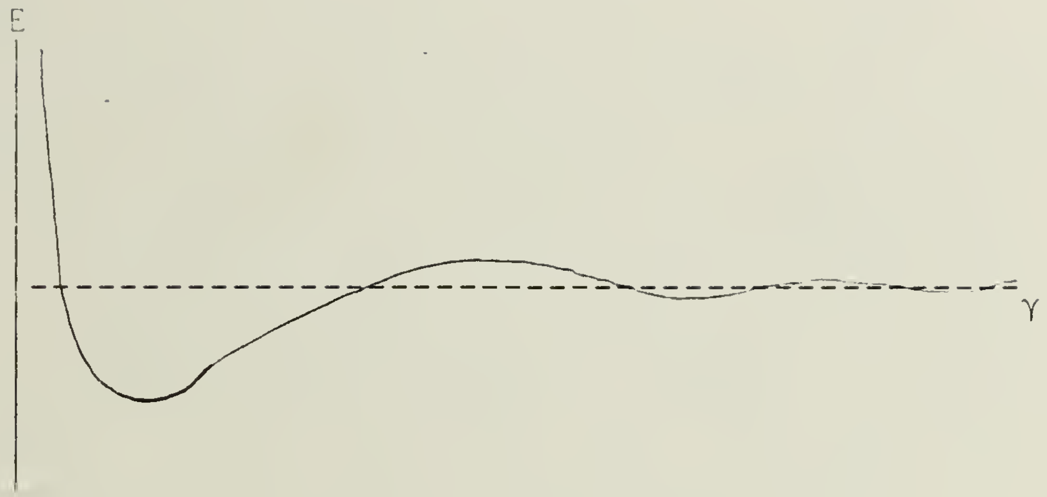


Figure 11. Energy Potential of a Multi-Body System.

An atom has a kinetic energy that is characteristic of the thermal energy of its surroundings. This motion is known as the thermal motion of the atom.

When the thermal motion and the interatomic potential of the crystal structure are combined, the atom's motion becomes a vibration. An atom oscillates about its site, always moving near its lattice position. Add a diffusing atom, in an interstitial site of the crystal, with its own vibration and its own potential energy and this interstitial atom will gain and lose energy as it vibrates between the lattice atoms. Figure 12 shows a qualitative graph of energy versus time of an atom's history while in an interstitial site.

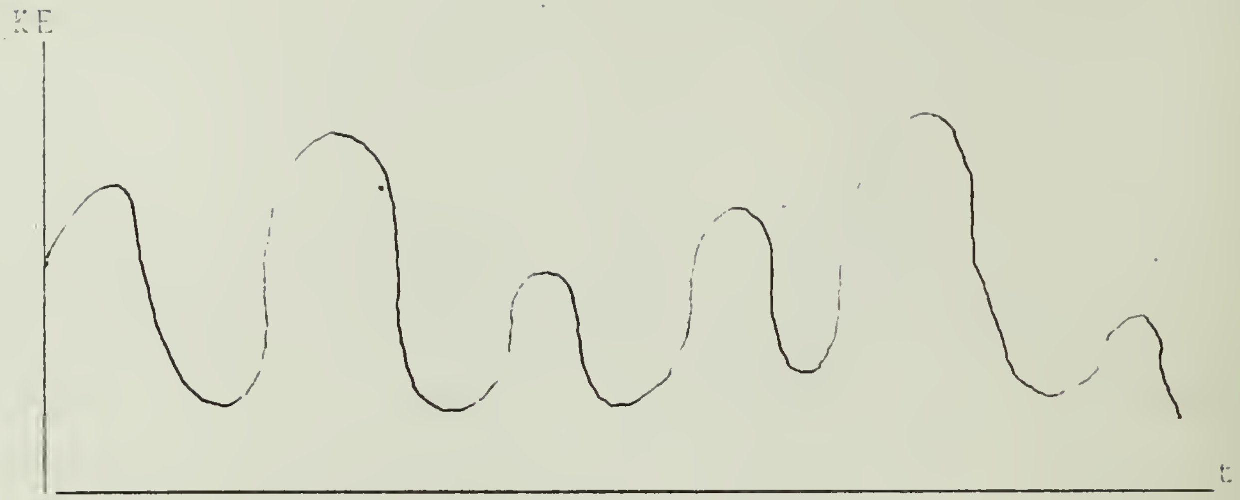


Figure 12. Time History of an Atom's Energy.

The potential energy of the atoms in a crystal structure sets up an energy barrier that a diffusing atom must overcome before it can move to one of its neighboring sites. This energy is known as the activation energy for diffusion. Figure 13 shows a qualitative graph of an atom's kinetic energy with respect to time, with the activation energy for a move indicated.

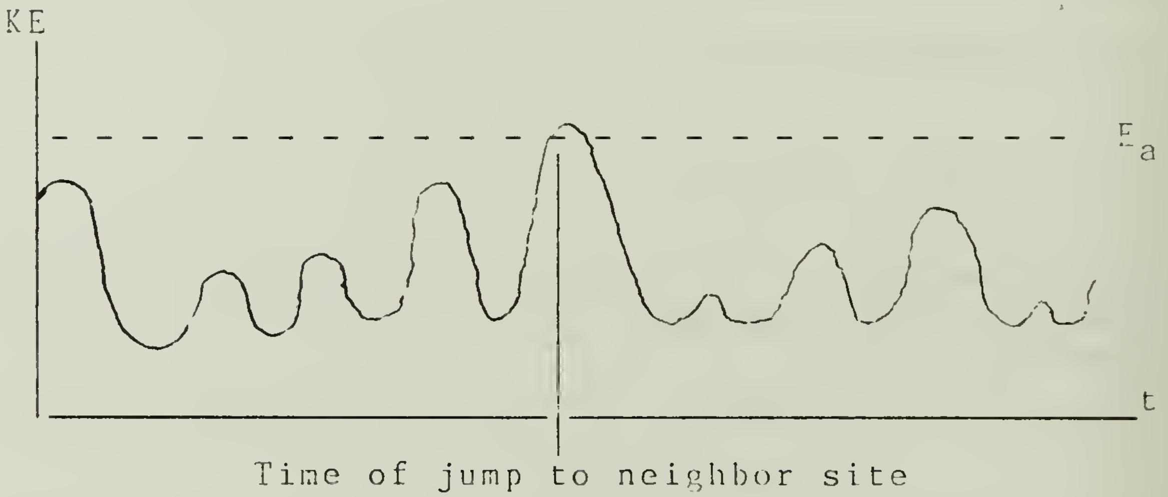


Figure13. History of Atom's Energy Noting Atom's Activation Energy.

D. RANDOM WALKS

Early in the study of diffusion mechanisms, the possibility of simulating the overall motion of diffusion was investigated. The method devised is still used today and is known as a Random Walk Process. A random walk in one dimension can be described as assigning a direction, either forward or backward, to the faces of a coin. When the coin is flipped, one step is made backward or forward according to the face of the coin displayed. Let R_n be the vector connecting the origin to the final position of the atom. The equation of R_n is then:

$$R_n = r_1 + r_2 + \ldots = \Sigma r_i \tag{6}$$

where r_i are the steps representing the individual jumps of the atom. If each direction of motion is equally likely and individual jumps are of equal magnitude, then the magnitude of R_n is

$$(R_n)^{1/2} = rn^{1/2}. \tag{7}$$

Now let a number of atoms, initially at x=0, each complete one dimensional random walks. Close to 95% of the atoms should be within + or - $\sqrt{R_n}$ of the origin.

E. COMPUTER SIMULATION OF DIFFUSION

Shortly after the development of the electronic computer Mayer and Ulam proposed that computers be used to simulate physical systems, Metropolis (1953), and King (1951) suggested that the Monte Carlo method of simulation (this method will be discussed in depth below) could be used to simulate the random walks of particles. The problem was that, at the time of King's paper, the computational capabilities of the machines in use were easily exceeded by Monte Carlo simulations. By 1961, computer technology was sufficiently advanced that Flinn and McManus were able to complete a Monte Carlo simulation of solid state diffusion. However, as late as 1970, there still had been no simulation of the diffusion processes based on the random walk theory. By 1970, it was evident that a new simulation process was required since the analytical processes of diffusion were unable to solve the topical diffusion problems evolving in the literature. In 1971 Bennett and Alder published the results of their simulation of diffusion by vacancies and divacancy random walks. Their paper marked the beginning of the random walk simulations of diffusion. Through the 1970's and 1980's, the field of Monte Carlo simulation of the diffusion process came into its own. Simulations have been used to test the correlation factors of the diffusion coefficient (Guy et al, 1977; Guy, 1978), the self diffusion of tracer atoms in an alloy (Bakker, 1976), and many other specific diffusion mechanisms (Murch, 1984).

F. CONCENTRATION DEPENDENT DIFFUSION COEFFICIENT

Concentration dependent diffusion was apparent early in the study of diffusion and the diffusion coefficient. C. Mantano (1933) made the first attempt to mathematically solve for the experimental results that showed the concentration dependence. Using the experimental results of metal-metal diffusion, he found that the concentration dependent diffusion coefficient would have the following form:

$$D(C) = (-1/2t)\ (dx/dC) \int x\,dC. \qquad (8)$$

This equation is known as the Matano solution for the diffusion coefficient.

Bowker and King (1978) attempted to develop a Monte Carlo simulation to determine the concentration dependent diffusion coefficient. Their first attempt used a two dimensional lattice gas model of a simple square plane and their results appeared to correspond with the experimental data of oxygen-tungston diffusion. However, a more detailed investigation of the simulation led to discovery of problems in the data collection of the concentration (Murch, 1981). Murch (1980) also tried to develop a simulation that could provide a concentration dependent diffusion coefficient (this simulation is discussed below), but his simulation showed no concentration dependence.

A recent paper by A.J. Slavin and P.P. Underhill (1984) attempted to prove that a concentration dependent diffusion coefficient could not be found using a random walk simulation of diffusion. However, a paper by R. Ghez and W.E. Langlois (1986) showed that a concentration dependent simulation can be developed if care is taken in the collection of the fluxes at the midpoint between the planes of the crystal.

II. OBJECTIVE

The primary objective of this thesis is to investigate the possibility that the diffusion coefficient is position dependent. The idea that such a dependence might exist is not new. In fact, a positional dependence has been observed in a crystal when there is a temperature gradient across the structure. It has always been accepted that when this gradient is absent, and the temperature is constant throughout the structure, the diffusion coefficient would be independent of the position of the diffusing atom. The idea of a position dependent diffusion coefficient was proposed by Collins (1985) as one way to test the assumption that the partition function, in Gibbsian statistical mechanics, does not divide into purely positional and velocity parts.

Though proposed early in the history of simulation, diffusion simulations did not actually begin until Bennett and Alder (1971) developed their program. They were studying persistence effects in hard sphere gases and used a simple Monte Carlo simulation to find the vacancy correlation factor in a two-dimensional fcc crystal. The first actual simulation of three dimensional diffusion was done by deBruin and Murch (1973). This simulation attempted to develop a method to determine the diffusion correlation factor. The diffusion correlation factor (f) is based on the Einstein diffusion coefficient method of determining the coefficient. Redefining Equation 5,

$$D = n\lambda^2 f/2t \tag{9}$$

the diffusion corellation factor can be defined as

$$f = \langle x^2\rangle/n\lambda^2 \tag{10}$$

where n is the number of jumps the diffusing particle takes, λ is the distance of each jump, and $\langle x^2\rangle$ is the mean-square displacement of the diffusing particle after time t (Le Claire, 1970). Early simulations of diffusion dealt with either diffusion of a vacancy, or self diffusion, and all were based on the Einstein diffusion equation (see Equation 5).

The first reported use of Fick's First Law in a diffusion simulation was by Guy et al. (1977) and Guy (1978) when the correlation factor was calculated. The method used in these investigations was to place an initial concentration of vacancies in a chemically homogeneous square planar structure and then allow the vacancies to move throughout the structure by a random walk process for a period of time. This method was found to have problems in that they had assumed that the atomic flux and the vacancy flux across the plane were different, and set up the simulation in that vein. In 1979 Murch and Thorn again tried to simulate the tracer correlation factor by this method. Their method was to set up a three-dimensional array with an inexhaustible reservoir of tracers or vacancies at one end (x=0), and at the other end (x=1) remove the tracers into a sink. The array was periodic in the y and z directions. From this simulation one could calculate the net flux across a plane, and from this calculate the correlation factor. In their paper, Murch and Thorn reported that this method gave as accurate answers as the Einstein method and was not so difficult to use.

Murch (1980) also used the Einstein method (Equation 5) to test the effect of diffusing atoms concentration on the chemical diffusion coefficient. This simulation was based on a three-dimensional simple cubic array. Within this array, he placed two reservoirs of atoms eight lattice units in from the x boundaries, each with a separate chemical potential. He then allowed the atoms to diffuse using a nearest-neighbor interaction, interstitial mechanism. He concluded from these investigations that the concentration did not affect the diffusion coefficient.

Fick's second law was not exploited in a diffusion simulation until 1978, when Bowker and King used it to study the diffusion coefficent. After initializing a lattice of vacancies, they placed a concentration of defects in one half of the lattice, and a different concentration in the other half. After a number of random steps of these defects, the concentration across the structure was counted plane by plane and a concentration profile was developed. From this profile the diffusion coefficient was derived numerically. Murch (1981) used the second law method but with the external reservoirs that he and Thorn (1979) developed for their first law investigation. He noted that if he used different data planes to develop his concentration profile he found different results. This led him to believe that the method had some problems with the development of the concentration gradient across the structure.

The development of these simulation techniques has proven that the diffusion coefficient can be studied using a digital computer. Each of these simulations used the premise that the coefficient is constant throughout the structure, but Murch (1980) suggested that these techniques could be used to see whether the diffusion coefficient has a positional dependence. Another limitation to all of these simulations is that the authors used either the Einstein method to determine the diffusion coefficient or Fick's first or second law. The simulations were never analyzed with more than one technique.

A comparison of the diffusion coefficient as calculated by Fick's first and second laws will be the primary method to determine the positional dependence in this thesis. If a positional dependence is recorded, a method other than

Fick's laws will be necessary to determine the diffusion coefficient. This method is known as the `Fokker-Plank' equation of diffusion:

$$\partial c/\partial t = -\partial /\partial x[A(x,t)C(x,t)] + 1/2\partial^2 /\partial x^2[D(x,t)C(x,t)] \quad (11)$$

In this Equation (11) the first element of the sum constitutes a drift factor where A(x,t) is a drift velocity, and the second element is the diffusion equation. The equation actually is giving the time evolution of the probability function, C(x,t). In the simulation this probability function is the concentration. If the time derivitive, spatial derivatives, and the functional form of the diffussion coefficient are known, the diffusion coefficient can be fit to the simulation data.

III. COMPUTER MODEL AND SIMULATION

A. PHYSICAL MODEL

A computer simulation model must mimic a physical model or experiment. In this simulation the physical model is a simple face-centered-cubic crystal membrane which is infinite in the x and y axis and very thin (less than 50 lattice planes) along the z axis. The mechanism will be interstitial diffusion. This means that the diffusing atoms are very small in comparison to the size of the lattice atoms. These diffusing atoms start in an inexhaustible reservoir outside the input plane of the crystal,and are completely absent from the volume outside the other side of the crystal at time equal to zero. At this starting time, the atoms begin to diffuse. After a period of time the distribution of diffusing atoms approach a steady state; in other words the transient effects on the concentration profile have ceased. At this time the flux across each plane in the crystal structure is checked and the concentration gradient is recorded. The diffusion coefficient is then found from these values. This thesis consists of a computer simulation of this physical system, which was designed to study the diffusion coefficient.

B. COMPUTER SIMULATION

The physical model is translated in such a way that the computer can follow the actions that take place. (This is known as computer simulation or computer modeling of a physical system.) Computer simulations can be separated into two families: continuous-time simulation and discrete-event simulation. Continuous-time simulations (also known as timestep simulations) simulate events which can be described by simultaneous equations. The equations describing events are allowed to proceed for a predeterminedperiod of time (a timestep), and then information is placed back into the equations which are to be solved numerically. Discrete-event simulations are used for physical systems which cannot be described by a mathematical function. The simulation runs from event to event rather then advancing a number of actions during a timestep.

Simulations can also be divided into types by the methods used to make decisions. If the simulation uses random numbers to make decisions, it is called a Monte Carlo simulation. This is because early simulations (those prior to computers) used devices, such as dice, to choose the random numbers that were used to make the decisions. Once simulations became more sophisticated, the community tried (in vain) to shift to the name `stochastic simulation', but the name `Monte Carlo' has stayed popular. In the other type of simulation, called `deterministic simulation', all decisions are made by rules set down prior to the actual running of the simulation program (Harrison,1985). The simulation used in this thesis is of the Monte Carlo type.

C. RANDOM WALK

The diffusion events are simulated by a method called a three dimensional random walk process. This process is simple in form and çan easily be set up. Consider a two dimensional example: Imagine that a marble is on a Chinese Checkers board. The marble has six neighbor sites to which it can be moved. A random walk of the marbles can be devised where each direction that the marble can move is assigned a number between one and six. A die is rolled and the marble is moved in the direction indicated by the number on the face of the die. This is an example of an uncorrelated two-dimensional random walk. Other forms of random walks are the `weighted' and `correlated' random walk. Returning to the Chinese Checker board example, if you move two marbles, one that is black and one that is white, with the same roll of the die, that is, if every time the die is rolled for the white marble you also move the black marble, then the two moves are connected and the random walk is correlated. An example of a weighted random walk can be seen by taking two dice and assigning one number to each of the four directions you want to have a iesser chance of the marble moving to, and assigning three numbers to each of the other two directions. This is a weighted random walk in the direction of interest. The random walk used in this thesis is a non-weighted, uncorrelated random walk.

D. COMPUTER CODE

The code for the computer simulation is divided into two separate programs, 'DIFSET' (for diffusion simulation set-up program) and 'DIFFUSE' (for diffusion simulation program). These programs combine to form a Monte Carlo simulation of an interstitial mechanism type of diffusion.

1. Difset

The program 'difset' can be separated into three parts, each of which intializes a table:

TABLE 3

TABLE OF DIFSET ARRAYS

Crystal site table

Nearest-neighbor table

Site vs plane table.

a. Crystal Set Up

The face-centered-cubic crystal used in this simulation is set up in a three dimensional array called CRYST3(I,J,K). The three dimensional array variables must meet certain limitations. These limitations are that the I and J variables have to be even numbers. There are no limitations on the K variable. The reason for these limitations is that the I and J axis in the simulated fcc crystal are set up to have periodic boundaries. The initial array is a perfect face-centered-cubic crystal with no defects or vacancies. An array position designates the center of a site and gives no indication of the size of an atom that might occupy the site. Atom size is not a factor in the simulation or in the random walk used, and therefore is not included in the set-up beyond the assumption that the diffusing atoms are small enough (relative to the lattice atoms in the crystal) to allow the interstitial mechanism of diffusion to occur. The set-up of the array CRYST3(I,J,K) is accomplished by placing the number two (2) in the array's register as an indication that a lattice atom is in that array position. To accomplish this, all the array positions where the I, J and K variables add to an odd number are assigned the number two (2). Table 4 lists examples of the initial value set up in `DIFSET' for the lattice positions.

TABLE 4

EXAMPLES OF ARRAY ENTRIES TO THE CRYST3 ARRAY.

CRYST3(3,1,1) = 2

CRYST3(2,2,1) = 2

CRYST3(2,1,2) = 2

CRYST3(5,2,4) = 2

The interstitial positions of the crystal are set up in the same way. Site indices that add to an even number are assigned a zero value to indicate that the interstitial postitions are empty. The crystal's first plane interstitial sites are filled by diffusing atoms. Therefore the array positions (even,odd,1) and (odd,even,1) are given the value of one (1) to indicate that these interstitial sites are filled with the diffusing atoms. The array is now set and an example plane is shown in Table 5.

TABLE 5

EXAMPLE OF THE FIRST TWO PLANES IN A 4X4XK CRYSTAL.

CRYST3(I,J,K) plane #1

2	1	2	1
1	2	1	2
2	1	2	1
1	2	1	2

CRYST3(I,J,K) plane #2

0	2	0	2
2	0	2	0
0	2	0	2
2	0	2	0

The three dimensional array that was set up is difficult for the computer to use, so a one dimensional array, CRYST(L), is set up to be equivalent to the CRYST3(I,J,K) three dimensional array. For methodology see code in Appendix B.

b. Nearest-Neighbor Table

The next major step of the 'DIFSET' program is to set up a nearest-neighbor table that will be used to simplify the random walk process. To review the diffusion by interstitial method, there are twelve interstitial sites which are nearest-neighbors to any other interstitial site. Therefore, a diffusing atom must move into one of these twelve nearest-neighbor sites. The array CRRNBR(L,R) (for crystal nearest-neighbor array) is set up such that the value held in the register is the L variable of another interstitial site in the CRYST(L) array. Thus if you want to know the twelve nearest-neighbors to CRYST(13), you would look them up in the CRNNBR(13,R) where the twelve R values are the twelve nearest-neighbor site L variables. The periodic boundary conditions for the I and J directions are set up in this array. The boundaries in these directions are made periodic by setting up the neighbor table to take an atom that leaves the boundary of the crystal on the left side, and replace it on the right side of the crystal as if the left and right side were neighbor planes. The edge sites of the K axis are the entry and exit positions of diffusing atoms and as such have special nearest-neighbor table values. The entry edge is simulated such that the plane is in contact with a infinite supply of the diffusing atoms, so the interstitial sites are always filled with the diffusing atoms. Another way of looking at this assumption physically is: if a diffusing atom leaves a surface interstitial position, another diffusing atom takes the interstitial position before the simulated timestep is completed. Therefore, the neighbor table is set up to contain only the four neighbors that are within the crystal. This means that a diffusing atom has the opportunity to enter the crystal only one third of the time. An example of this table is located in Appendix F.

c. Site vs Plane Table

The final item that the program 'DIFSET' develops is the lattice site plane table. This array ZPLN(L) (representing the plane # in the z direction) correlates the site number L to the plane number which is contained in ZPLN(L).

The information developed in 'DIFSET' is then placed into an output file to be used as input to the other program 'DIFFUSE'.

2. Diffuse

'DIFFUSE' is the actual simulation program that models the diffusion process. This is done using a Monte Carlo timestep simulation. At each timestep, the simulation looks at each crystal site. If it contains a diffusing atom the simulation then looks at a neighbor site selected at random. If the neighbor site is empty the diffusing atom is moved into that neighbor site. In the simulation, the process is done with two tables of random numbers. At the beginning of each timestep, a random number generator is called to make a table of random numbers, one for each site in the CRYST(L) array. The random number is uniform, 1 to LMAX, where LMAX is the total number of sites within the array. The sites are characterized in this way, so the simulation can be used later to study diffusion by the vacancy mechanism with little modification. Another table of random numbers uniform 1 to 12 is set up. Again the size of the table is equal to the number of sites in the array CRYST(L). As a time saver, during the first few timesteps the number of random numbers generated is equal to the number of sites in one plane multiplied by the timestep, and only those first planes are used in the simulation.

After the random numbers for the timestep are assigned to their tables, the simulation calls the first random number (from the uniform 1 to LMAX set), sets that number equal to L and looks to see if CRYST(L) contains a diffusing atom. If the site does not contain a diffusing atom, another random number is selected from the first table. If there is a diffusing atom in the site, a random number is selected from the second table. This number is set equal to R and the nearest neighbor table is used to find the value in CRMNBR(L,R). This value is then set equal to L1 and if CRYST(L1) equals zero the diffusing atom is moved into the new location. This process continues until all LMAX numbers have been tested.

During each of these moves a number of factors required for the numerical solution of the diffusion coefficient are calculated.

a. Concentration

The concentration of atoms in each plane is determined at the beginning of each timestep by looking at every site in the array. If the register contains a one (a diffusing atom), then the concentration in that plane is increased by setting K = ZPLN(L), and adding one to the value already in the array CONS(TMSTP,K). The concentration is then divided by the number PLNMAX, which is the total volume of a single plane. This puts the concentration into units of particles per unit volume. These units are needed for use in Fick's law. This value is stored in the array CVOL(TMSTP,K).

b. Flux

The flux density through each plane is determined when an atom is being moved from the old site to its new site. Since the flux density is actually between two planes, by definition the area between plane K(1) on the left and K(2) on the right will be considered part of plane K(1). This means that if an atom diffuses from K(1) into K(2) then the flux of K(1) has 1/PLNMAX added to it.

However, if the atom moves from K(2) back to K(1) the flux of K(1) is decreased by 1/PLNMAX. Thus FLUX(TMSTP,K(1))) = FLUX(TMSTP,K(1)) + 1/PLNMAX if the atom moves from K(1) to K(2) and FLUX(TMSTP,K(1)) = FLUX(TMSTP,K(1)) -1/PLNMAX if it moves from K(2) to K(1).

c. First Derivative

The first derivative of the concentration with respect to position is calculated by subtracting the concentration per unit volume of the plane after the plane where the derivative is required, from the plane preceding the plane where the derivative is required. For example DERC1(TMSTP,K) = CVOL(TMSTP,K+1) - CVOL(TMSTP,K-1). The units for the concentration gradient, as stated above, are particles per inter-planar distance to the fourth power. With these units the change in the position is 1 and the mathematics is simplified.

d. Second Derivative

The second derivative with respect to the position is obtained in the same manner, with the exception that the 1st derivitve is used rather than the planar concentration. DERC2(TMSTP,K) = DERC1(TMSTP,K+1) - DERC1(TMSTP,K-1). The units for the second derivative are particles per inter-planar distance to the fifth power.

e. Time Derivative

The time derivatives are derived in the same manner as the positional derivative. The concentration of the plane at the beginning of the timestep is subtracted from the concentration at the end; DERT(TMSTP,K) = CVOL(TMSTP+1,K) - CVOL(TMSTP,K). The units for this factor are particles per timestep per inter-planar distance cubed.

f. Averaging

After a pre-determined number of timesteps have been completed, the arrays containing the information are averaged. This gives another set of arrays that are used to derive the diffusion coefficient. The single timestep arrays are then reset to zero, and another set of timesteps are begun.

g. Diffusion Coefficient

In this simulation the diffusion coefficient is derived by two methods so that the results can be compared and the positional dependence of the coefficient may be more easily seen. The first method uses Fick's first law. Having found the average flux over a number of timesteps, AFLUX(ATMSTP,K), and the average first derivative, ADERC1(ATMSTP,K), the diffusion coefficient can be found by dividing the average flux by the negative of the derivative, DIFCO1(ATMSTP,K)= -AFLUX(ATMSTP,K)/ADERC1(ATMSTP,K). The diffusion coefficient is not determined for every timestep because, at times, the first derivative for a single timestep is zero and the resulting diffusion coefficient is infinite. This result would skew the averaged diffusion coefficient.

The second method of determining the diffusion coefficient is based on Fick's second law of diffusion. More specifically, the second law holds that the diffusion coefficient is constant with respect to position, $\partial c/\partial t = D\partial^2 c/\partial x^2$. The diffusion coefficient is then determined by dividing the time derivative of the concentration by the second positional derivative, DIFCO2(ATMSTP,K) = ADERT(ATMSTP,K)/ADERC2(ATMSTP,K)). Averages are used here for the same reason as described for the first method.

IV. RESULTS

A. PROOF OF SIMULATION'S PHYSICAL EQUIVALENCE

The proof of the simulation's ability to model the physical world can be seen in three ways. First the simulation adheres to the conservation of matter. Second, the diffusion coefficient that is developed from the simulation's data lies within the limits of the experimentally found values for the diffusion coefficient. Third, the time average of the concentration profile approaches the infinite time result predicted for the concentration.

1. Conservation of Matter

The conservation of matter and Fick's second law (Equation 3) are equivalent. Therefore, if Equation 3 holds in the simulation then the conservation of matter holds by definition. To test that this requirement is met at all times during the simulation, the first derivative of the flux density was determined and compared to the time derivative of the concentration during one run of the simulation. It was found that the conservation of matter held for all time in the simulation.

2. Comparison of the Diffusion Coefficient

The diffusion coefficient developed from the simulation data is in units of inter-planar distance squared per timestep. In order to compare the diffusion coefficient found by experiment with the coefficient developed from the simulation, the lattice spacing of the base material, the temperature of the system, the mass of the diffusing particle, and the force constant of the harmonic oscillator approximation to the potential function between the diffusing atom and the base material must be provided. As an example, the diffusion coefficient for carbon diffusing through fcc iron at 910 degrees Celsius has been experimentally determined to be 1.91 x 10^{-7} cm^2/s (Askeland, 1984). The lattice unit for fcc iron is 3.589 x 10^{-8} cm (Askeland, 1984) , the force constant for carbon-iron is in the range 2.65 x 10^{8} to 17.73 x 10^{8} dynes/cm (A.J. Gordon, 1972), and the atomic weight of carbon is 12.011 gr/mol. From these constants the experimentally determined diffusion coefficient listed above is transformed into a range of values that fall between 0.23 and 0.61 inter-planar distance squared per timestep. The diffusion coefficient, values determined by the time averaging over 100 timesteps, fall between 0.19 and 0.83 inter-planar distance squared per timestep. The values for the coefficient determined by the simulation's time averaging over 1000 timesteps is between 0.26 and 0.42 inter-planar distance squared per timestep. This indicates that the simulation is determining the diffusion coefficient within acceptable limits.

3. Concentration Profile

The concentration profile across a membrane should proceedtoa linear slope across the planes of the crystal structure. The simulation 'DIFFUSE' begins to approach the correct form at the 1-1000 time average of the concentration (see figure 14). It then proceeds to oscillate about the infinite time profile result. (see figure 15-17). This oscillation about the infinite time line indicates that the simulation reaches an approximate steady-state configuration. To test this hypothesis, the 19001-20000 time average profile was statistically checked against the linear infinite fit. The statistical method used was a linear regression of the data to the infinite result. The regressionresults gave a probability of 0.999 that the data does fit the linear slope of the infinite time profile. The same approximate result was found for the 14901-15000 and the 60901-61000 time average concentration profiles. The sameregression was fit, using the data averaged over 19901-20000timesteps, found that the probability that its concentration profile could be fit to the infinite time profile was 0.953.

Because of the concentration profile's closeness of fit to the infinite time profile, the apparent derivation of the actual diffusion coefficient by the simulation, and the simulation's ability to obey the conservation of matter,

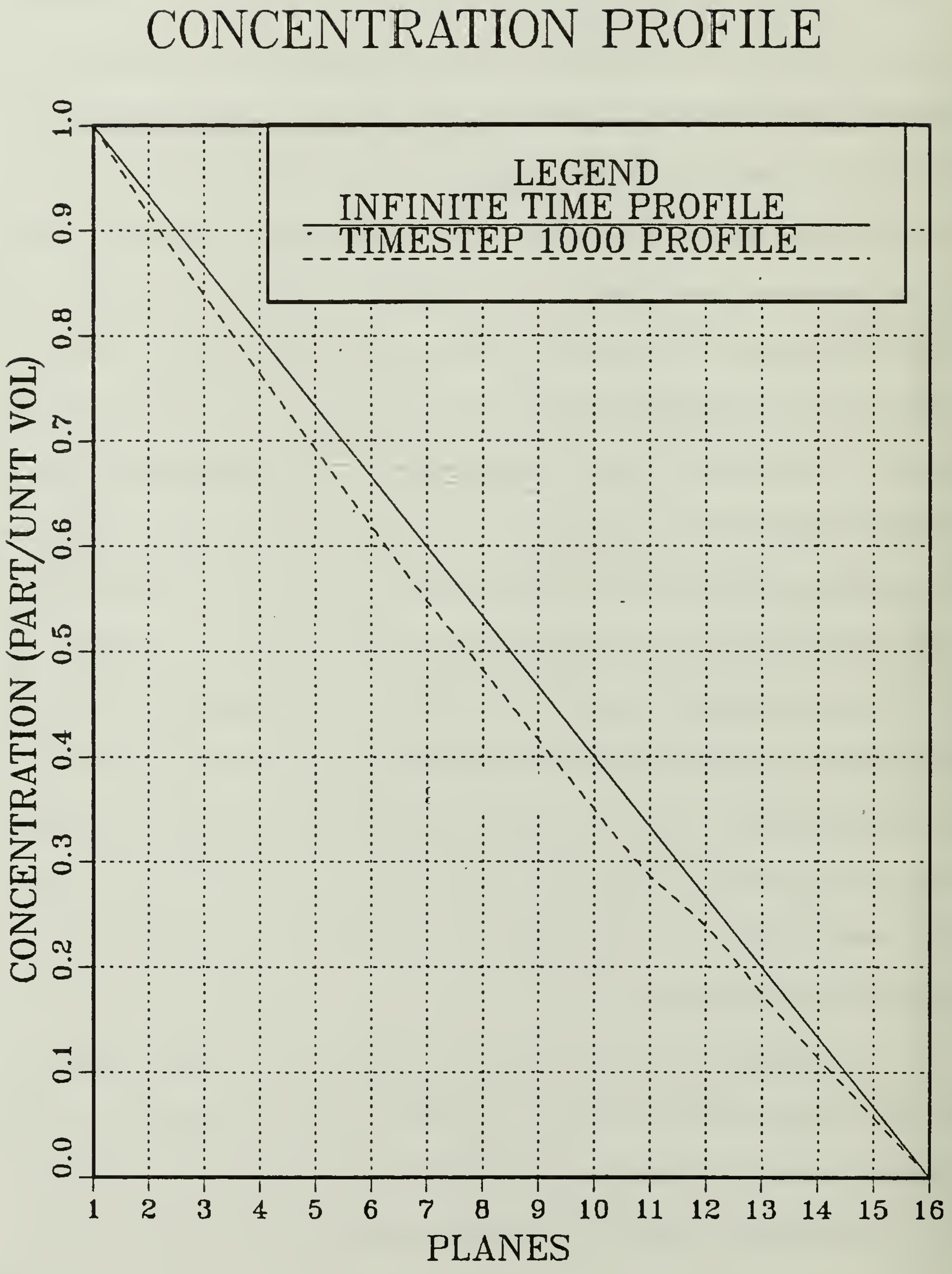

CONCENTRATION PROFILE
LEGEND
INFINITE TIME PROFILE
TIMESTEP 1000 PROFILE
CONCENTRATION (PART/UNIT VOL)
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
PLANES

Figure 14.

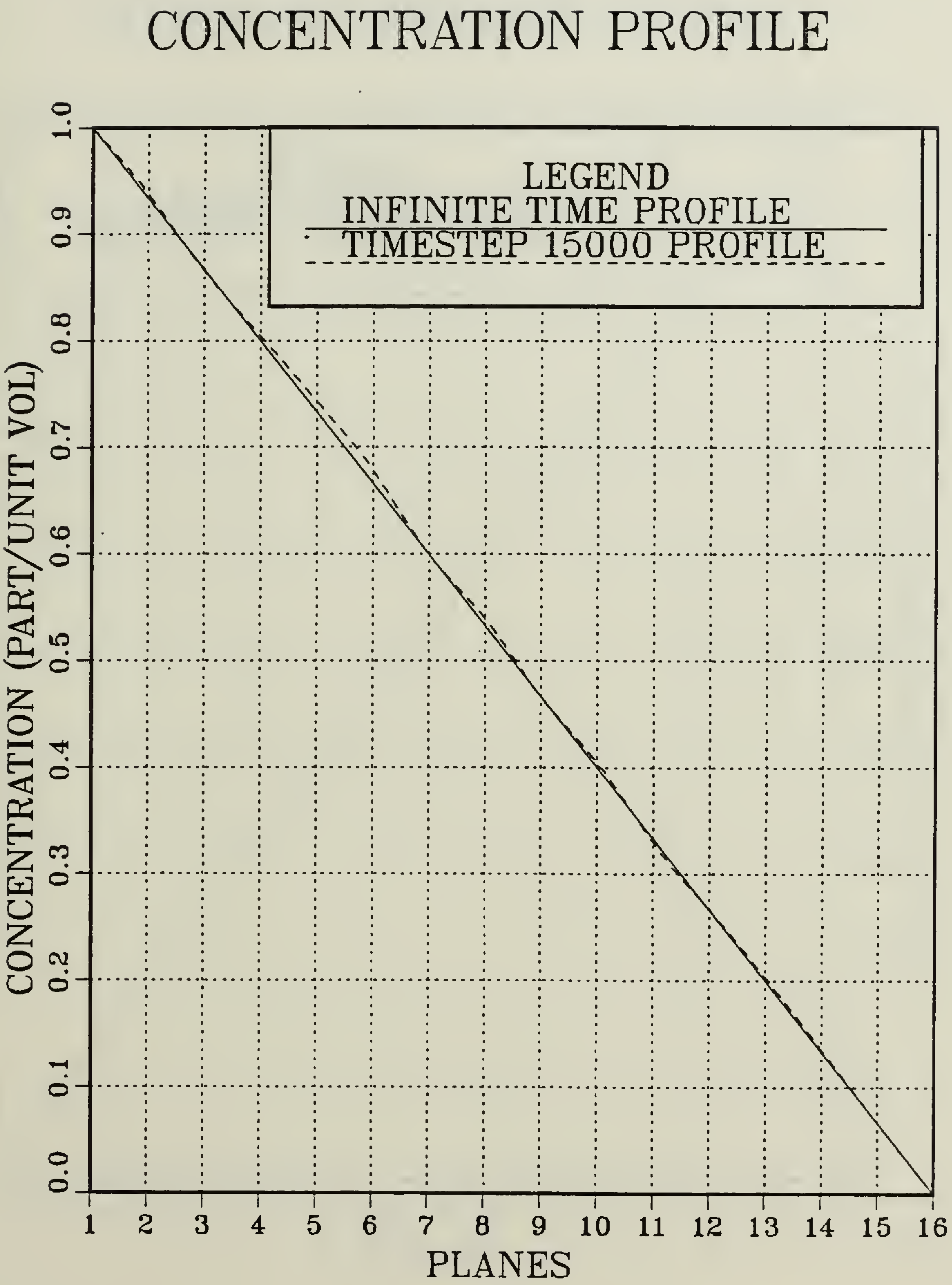

CONCENTRATION PROFILE
LEGEND
INFINITE TIME PROFILE
TIMESTEP 15000 PROFILE
CONCENTRATION (PART/UNIT VOL)
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
PLANES

Figure 15.

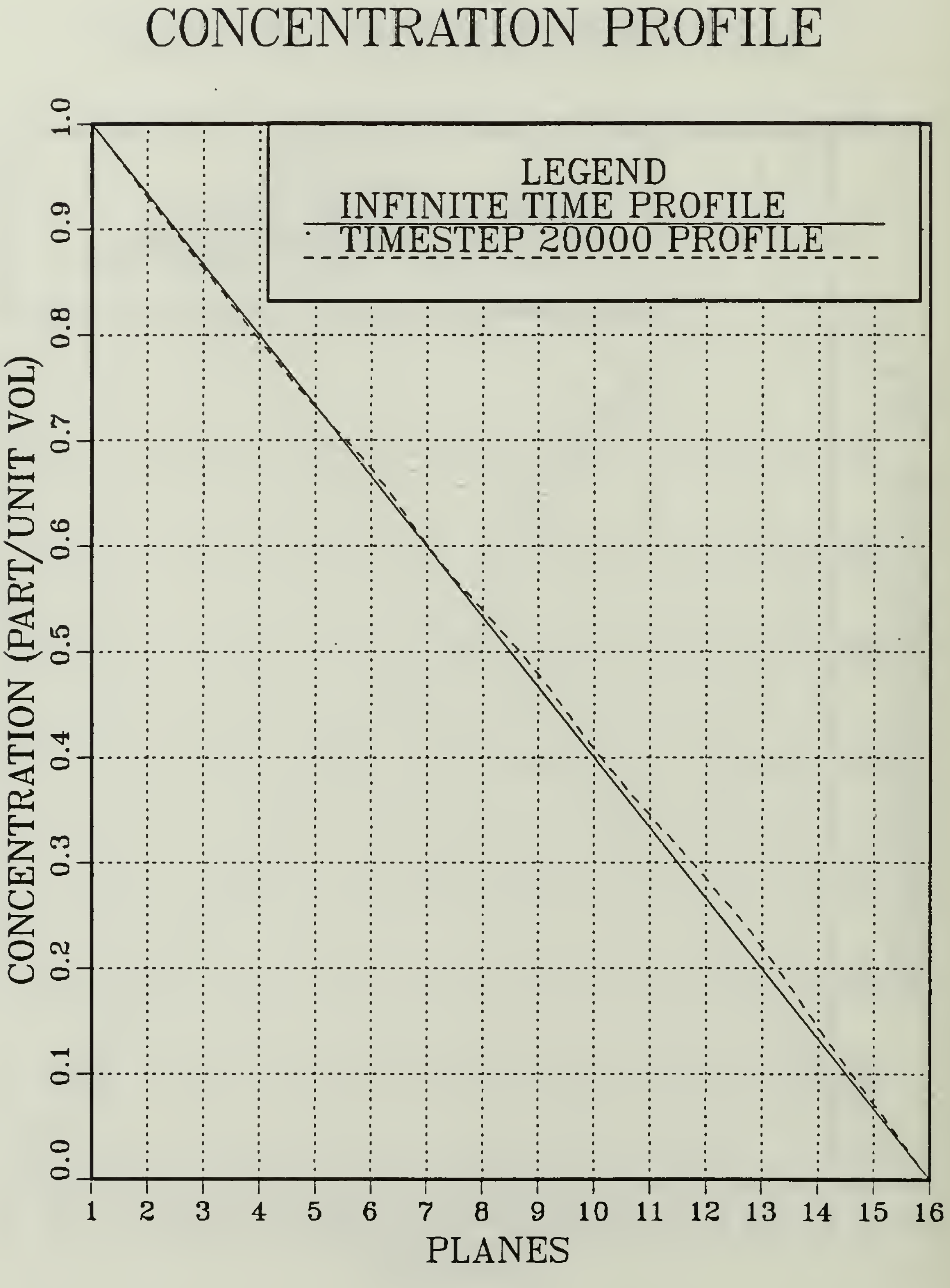

CONCENTRATION PROFILE
LEGEND
INFINITE TIME PROFILE
TIMESTEP 20000 PROFILE
CONCENTRATION (PART/UNIT VOL)
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
PLANES

Figure 16.

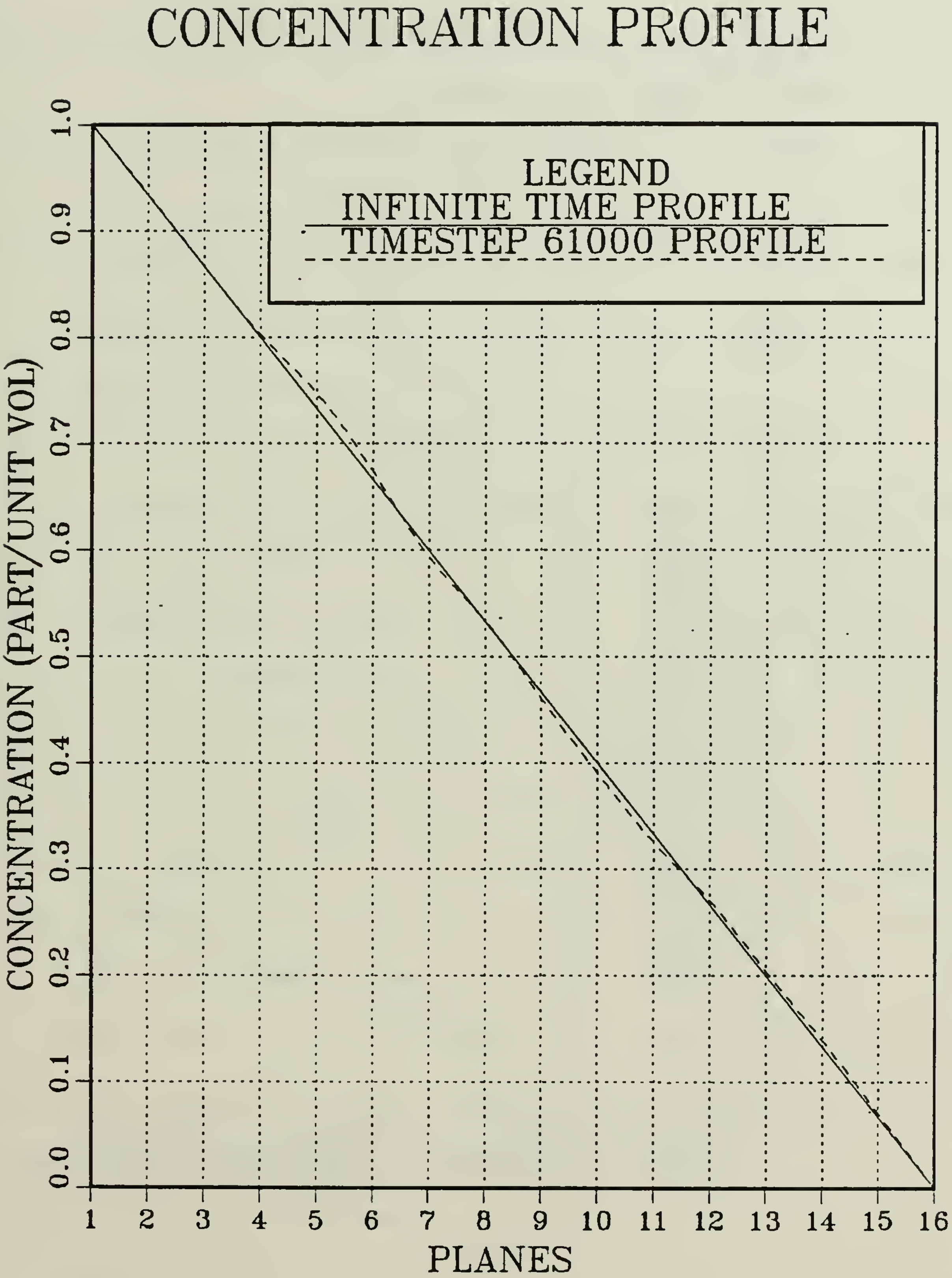

CONCENTRATION PROFILE
LEGEND
INFINITE TIME PROFILE
TIMESTEP 61000 PROFILE
CONCENTRATION (PART/UNIT VOL)
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
PLANES

Figure 17.

it is reasonable to assert that the simulation is acting as as an approximate physical model of the interstitial diffusion mechanism through a fcc crystal.

B. CALCULATION OF THE DIFFUSION COEFFICIENT

1. Fick's First Law Method

'DIFFUSE' initially develops the diffusion coefficient by using the Fick's first law method. This method takes the time average flux and divides it by the time average first derivative of the concentration. By this method, the diffusion coefficient shows what appears to be an oscillation across the planes (see Figures 18 & 19). The cause of these oscillations could be the result of either noise in the simulation, the data fitting a horizontal line across the plane, or a positional dependence on the diffusion coefficient. To test whether the data fit a horizontal line across the plane, a linear regression was fit to the data. The results of the fit are plotted in Figures 18 & 19. The linear correlation factor for the time average gives a probabillity of 20 - 30 per cent that the data fits a line of constant slope and less then 20 percent probability that the data fits a horizontal line across the planes.

To investigate the chance that the oscillation was noise and not a positional dependence on the diffusion coefficient, the diffusion coeficient data as a function of

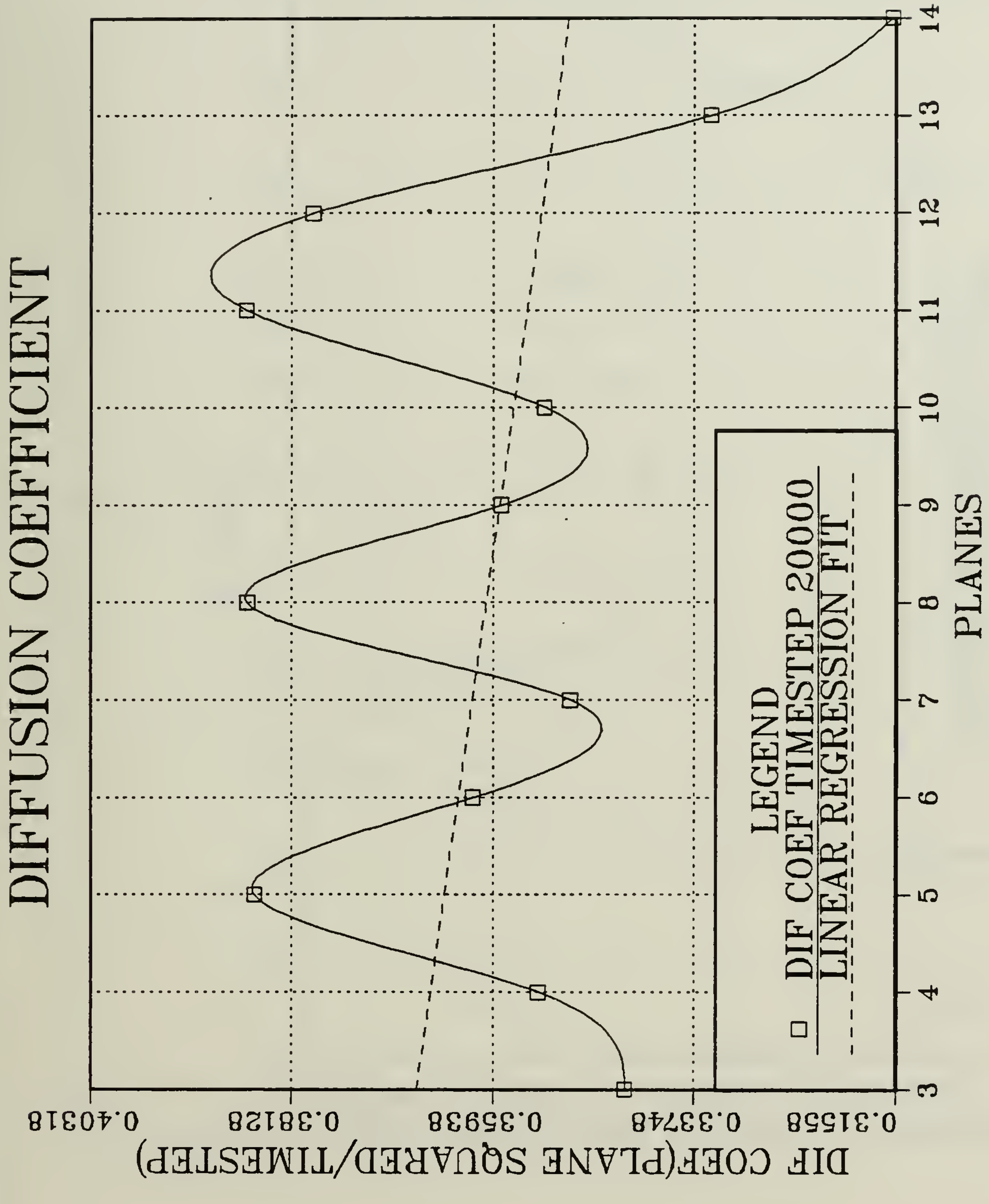

DIFFUSION COEFFICIENT
DIF COEF(PLANE SQUARED/TIMESTEP)
0.31558
0.33748
0.35938
0.38128
0.40318
PLANES
3
4
5
6
7
8
9
10
11
12
13
14
LEGEND
DIF COEF TIMESTEP 20000
LINEAR REGRESSION FIT

Figure 18.

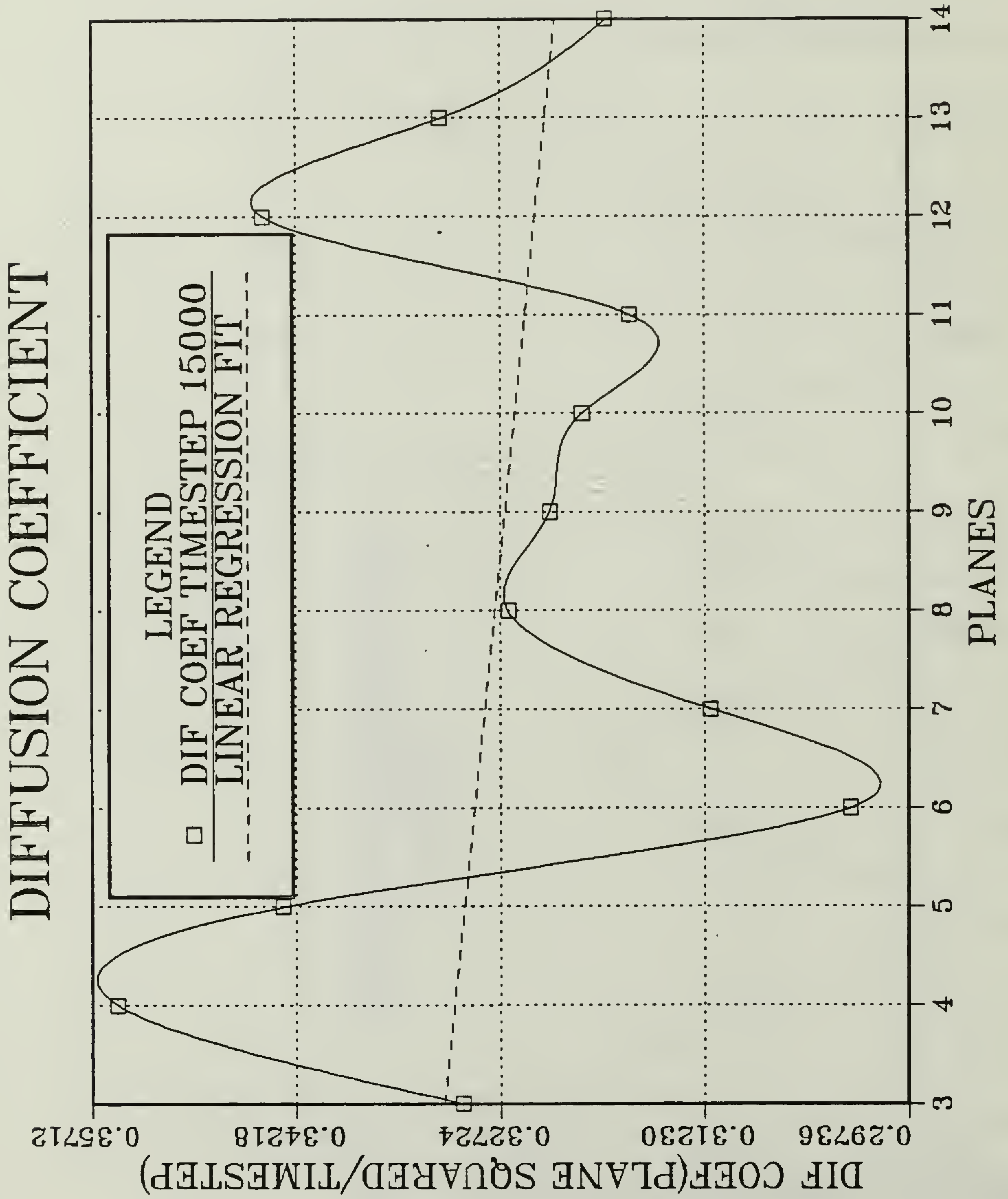

DIFFUSION COEFFICIENT
LEGEND
□ DIF COEF TIMESTEP 15000
LINEAR REGRESSION FIT
DIF COEF(PLANE SQUARED/TIMESTEP)
0.35712
0.34218
0.32724
0.31230
0.29736
PLANES
3
4
5
6
7
8
9
10
11
12
13
14

Figure 19.

It is apparent from the two tables that the results of the two methods are drastically different. Unlike the first method data the second law data is not equally distributed aboutits mean and all the data are not within two standard deviations of the mean. The means of the two methods are also not within two standard deviations of each other. Therefore, the diffusion coefficient may be dependent on the position of the diffusing atom. The next step in the test of the hypothesis is to find the first derivative of the diffusion coefficient and compare the time derivative of the concentration and the result of Fick's second law with the non-linear diffusion coefficient (Equation 3a). The results of this comparison can be seen in Tables 8 and 9.

TABLE 8.

TABLE OF SHORT TIME-AVERAGED TIME DERIVATIVES

Plane	Time Derivative	Fick's Second Law
3	.00056	-.02396
4	-.00056	-.00320
5	-.00222	-.00199
6	-.00111	-.00481
7	.00000	-.00173

NOTE: Table of time derivatives from simulation and derived by Equation 3a using the averages over the 19901-20000 timesteps.

TABLE 9

TABLE OF LONG TIME-AVERAGED TIME DERIVATIVES

Plane	Time Derivative	Fick's Second Law
3	.00017	-.02539
4	.00000	-.00142
5	-.00006	-.00027
6	-.00011	.00104
7	-.00011	-.00107

NOTE: Table of time derivatives from simulation and derived by Equation 3a using the averages over the 19001-20000 timesteps.

The results of this comparison show that the non-linear form of Fick's second law does not agree with the simulation results of the time derivative of the concentration. This suggests that Fick's first law, taken literally, may not be the best method to fit the simulation data to a diffusion coefficient. It may be the diffusion coefficient should be within the partial derivative.

3. Fokker-Plank Method

The last method used to search for a positional dependence was to attempt to fit a function to the diffusion coefficient and solve the Fokker-Plank equation (Equation 13) with this function and the results of the simulation. The following functional form was assumed, based on the diffusion coefficient resulting from the Fick's first law method:

$$D(x,t) = A + B\sin(Cx + Dt + E) \qquad (12)$$

Using this equation and the results from the simulation, the Fokker-Plank equation takes the following form:

$$\partial C/\partial t = B\sin(Cx+Dt+E)[0.5C^2(CVOL)+C(DERC1)=.5(DERC2)] + A/2 \qquad (13)$$

The IMSL library subroutine ZXSSQ was used to evaluate the Fokker-Plank equation. ZXSSQ finds the minimum of the sum of squares of M functions in N variables using a finite difference Levenberg-Marquardt algorithm. In the simulation the N variables are the A, B, C, D, and E variables of the diffusion coefficient function, and the M functions are Equation 13 evaluated at different timesteps from the simulation.

It was determined that the best fit for the coefficients A, B, C, D and E was when they all equaled zero. This result indicated that the oscillations in the diffusion coefficient were probably based on the numerical noise in the system.

V. CONCLUSIONS AND RECOMMENDATIONS

The data from the simulation gave no clear determination, one way or the other, about the positional dependence of the diffusion coefficient. The simulation should be further improved to make a specific determination of the positional dependence of the diffusion coefficient. The following improvements to the simulation should give it the ability to make the determination of the dependence.

First, the time averaging of the flux, concentration, and derivatives may have averaged out any real positional dependence. Therefore, the number of timesteps the simulation averages over should be shortened, or deleted altogether. Instead, the few times that the derivatives are zero should be ignored and the diffusion coefficient should be inspected at all other times and planes.

Second, the size of the array should be increased. The size used in this thesis, 6 X 6 X 16, may not have been large enough. It appears that with this size a single particle change in a plane leads to large changes in the parameters determined by the simulation. The array should be increased in size to at least a 14 X 14 X 50 array. This size should make a single particle change in a plane equal a one percent change in the determined factors of the simulation.

Third, the functional form of the diffusion coefficient, and the initial values for the analysis of the unkown variables for the Fokker-Plank equation need to be investigated further.

A major result of this thesis is its improvement in the bookeeping ability of diffusion simulations. 'DIFFUSE' uses arrays to keep track of values that are needed to simulate a more sophistacated model of diffusion. These arrays keep track of the nearest-neighbors for the diffusing atoms, and the planes that the atoms are in. This allows the simulation to be written more simply than earlier simulations and allows the more sophisticated modeling of the face-centered-cubic crystals to be simulated.

The simulation's improvements to the method by which the diffusion process is modeled, along with these suggested improvements to the simulation, should allow the simulation to determine if the diffusion coefficient has a positional dependence.

APPENDIX A

VARIABLE GLOSSARY

The following glossary of variable names is set up to aid in using the computer programs `DIFSET' and `DIFFUSE'. The glossary consists of three columns, the name of the variable, the type of variable and the physical meaning of the variable.

VARIABLE NAME	VARIABLE TYPE	PHYSICAL MEANING
ACONS	2-D ARRAY	AVERAGE CONCENTRATION AT EACH PLANE.
ACVOL	2-D ARRAY	AVERAGE CONCENTRATION PER UNIT VOL. AT EACH PLANE.
ADERC1	2-D ARRAY	AVERAGE 1ST DERIVATIVE OF THE CONCENTRATION AT EACH PLANE.
ADERC2	2-D ARRAY	AVERAGE 2ND DERIVATIVE AT EACH PLANE.
ADERT	2-D ARRAY	AVERAGE TIME DERIVATIVE OF THE CONCENTRATION AT EACH PLANE.
AFLUX	2-D ARRAY	AVERAGE FLUX OF EACH PLANE.
ATMSTP	ARGUMENT	USED AS 1ST ARGUMENT IN ALL THE AVERAGED ARRAYS.
AT1	PARAMETER	NUMBER OF TIMESTEPS FOR THE AVERAGED ARRAYS.
AVGTM	PARAMETER	NUMBER OF TIMESTEPS TO BE AVERAGED OVER.
AVGTM1	PARAMETER	PARAMETER USED TO HOLD LAST 100 TIMESTEPS IN MEMORY.

VARIABLE NAME	VARIABLE TYPE	PHYSICAL MEANING
CONS	2-D ARRAY	CONCENTRATION FOR EACH PLANE AND TIMESTEP.
CVOL	2-D ARRAY	CONCENTRATION PER UNIT VOL AT EACH PLANE AND TIMESTEP.
CRNNBR	2-D ARRAY	NEAREST-NEIGHBOR TABLE.
CRYST3	3-D ARRAY	THREE DIMENSIONAL ARRAY OF FCC CRYSTAL.
CRYST	1-D ARRAY	ONE-DIMENSIONAL EQUIVALENT TO CRYST3.
DERC1	2-D ARRAY	1ST DERIVATIVE OF THE CONCENTRATION AT EACH PLANE.
DERC2	2-D ARRAY	2ND DERIVATIVE AT EACH PLANE.
DERT	2-D ARRAY	TIME DEIVATIVE AT EACH PLANE.
DIFCO1	2-D ARRAY	DIFFUSION COEFFICIENT AT EACH PLANE, DETERMINED BY FICK'S FIRST LAW.
DIFCO2	2-D ARRAY	DIFFUSION COEFFICIENT DETERMINED BY FICK'S SECOND LAW.
FLUX	2-D ARRAY	FLUX ACROSS EACH PLANE AND TIMESTEP.

VARIABLE NAME	VARIABLE TYPE	PHYSICAL MEANING
IMAX	PARAMETER	NUMBER OF PLANES IN THE X DIRECTION.
IMAX1	PARAMETER	NUMBER ONE LESS THAN IMAX.
ISEED1	PARAMETER	RANDOM NUMBER SEED.
ISEED2	PARAMETER	RANDOM NUMBER SEED.
JMAX	PARAMETER	NUMBER OF PLANES IN Y DIRECTION.
JMAX1	PARAMETER	ONE LESS THAN JMAX.
K	ARGUMENT	ARGUMENT TO ALL 2-DIMENSIONAL ARRAYS. REPRESENTS THE PLANE.
KMAX	PARAMETER	NUMBER OF PLANES IN THE Z DIRECTION.
KMAX1	PARAMETER	ONE LESS THAN KMAX.
L	ARGUMENT	ARGUMENT FOR ARRAYS ZPLN AND CRYST.
LMAX	PARAMETER	NUMBER OF SITES IN CRYST.
LMAX1	PARAMETER	LMAX-PLMAX. ONE PLANE LESS THAN ALL THE SITES IN CRYST.
LRDM	ARGUMENT	ARGUMENT OF THE RDM1 AND RDM2 ARRAYS.

VARIABLE NAME	VARIABLE TYPE	PHYSICAL MEANING
L1	ARGUMENT	TEMPORARY ARGUMENT FOR CRYSTL TO FIND NEAREST NEIGHBOR.
MAVGTM	PARAMETER	LARGEST NUMBER OF THE AVERAGED TIMESTEPS.
NTMSTP	PARAMETER	LARGEST NUMBER OF TIMESTEPS.
OUTPUT	LOGICAL	DETERMINES AMOUNT OF OUTPUT THE PROGRAM GIVES.
RDM1	1-D ARRAY	ARRAY OF RANDOM NUMBERS USED FOR PICKING CRYST SITE TO BE LOOKED AT.
RDM2	1-D ARRAY	ARRAY OF RANDOM NUMBERS USED TO PICK NEAREST-NEIGHBOR TO BE LOOKED AT.
TMSTP	ARGUMENT	ARGUMENT FOR THE ARRAYS FLCK,DERC1,DERC2 AND DERT.
T1	PARAMETER	FIRST TIMESTEP NUMBER OF PROGRAM.
ZPLN	1-D ARRAY	ARRAY OF SITES VS PLANES.

APPENDIX D

'DIFSET' CODE LISTING

The following is the computer code for the set-up program 'DIFSET'. This code develops the input for the program 'DIFFUSE'.

```
C                              'DIFSET'
C                          MARK R. POLNASZEK
C                              JUNE 1986
C
C
C   'DIFSET' WAS DESIGNED TO INTIALIZE THREE ARRAYS THAT ARE NEEDED
C   IN THE PROGRAM 'DIFFUSE'.  THESE ARRAYS CRYST(L), CRNNBR(L,R),
C   AND ZPLN(L) ARE USED IN 'DIFFUSE' TO SPEED UP THE CALCULATIONS
C   OF THE DIFFUSION PROCESS.
C
C
C   THE FOLLOWING DEFINES THE VARIABLES OF THE PROGRAM AS REAL OR
C   INTEGER VALUES, AND DIMENSIONS THE ARRAYS' MEMORY SIZE.
C
C
00020  INTEGER PLNMAX,CRYST(2000), CRYST3(10,10,20),CRNNBR(2000,12),R,
     1      ZPLN(2000), ISEED1, ISEED2, FLUX(2000), CONS(2000),MTMSTP,
     2      T1,AVGTM,PLSTM,OUTPUT

C
C
C   THE FOLLOWING READS IN THE INPUT TO THE PROGRAM AND ECHOS IT
C   BACK TO AN OUTPUT FILE.
C
C
       READ (5, *)IMAX, JMAX, KMAX,MTMSTP,T1,ISEED1,ISEED2,AVGTM,PLSTM,
     1            OUTPUT
       WRITE (6,26) IMAX, JMAX, KMAX,MTMSTP,T1,AVGTM
00026  FORMAT ('1',6I8)
       WRITE (6,28) ISEED1,ISEED2
00028  FORMAT ('1',2I16)
C
C
C   THE CODE BELOW INITIALIZES CONSTANTS THAT ARE USED THROUGH OUT
C   THE PROGRAM.
C
C
00040  PLNMAX = IMAX * JMAX
00050  LMAX = KMAX * PLNMAX
       PRINT 52, PLNMAX, LMAX
00052  FORMAT ('1',2I8)
C
C
C   THE FOLOWING DO LOOPS SET UP THE FIRST ARRAY ZPLN(L).  THIS ARRAY
C   HOLDS THE VALUE OF THE PLANE NUMBER FOR THE GIVEN SITE NUMBER,L.
C
C
                     DO 3192 L = 1,LMAX
                     ZPLN(L)=(INT (L/PLNMAX))+1
                     DO 3191 N = PLNMAX,LMAX,PLNMAX
                     ZPLN(N)= N/PLNMAX
03191                CONTINUE
03192                CONTINUE
C
C
C   THE FOLOWING DO LOOPS ARE USED TO SET UP THE THREE DIMENSIONAL
C   ARRAY CRYST3(I,J,K).  THIS ARRAY HOLDS THE VALUE OF THE TYPE
C   OF ATOM IN THE CRYSTAL SITE OR THE FACT THE SITE IS EMPTY.
C
C
00060  DO 102 K = 2,KMAX,2
00062     DO 70 J = 1,JMAX,2
00064        DO 68 I = 1,IMAX,2
00066          CRYST3(I,J,K) = 0
00068        CONTINUE
00070     CONTINUE
00072     DO 80 J = 2,JMAX,2
00074        DO 78 I = 2,IMAX,2
```

```
00076         CRYST3(I,J,K) = 0
00078       CONTINUE
00080     CONTINUE
00082     DO 90 J = 2,JMAX,2
00084       DO.88 I = 1,IMAX,2
00086         CRYST3(I,J,K) = 2
00088       CONTINUE
00090     CONTINUE
00092     DO 100 J = 1,JMAX,2
00094       DO 98 I = 2,IMAX,2
00096         CRYST3(I,J,K) = 2
00098       CONTINUE
00100     CONTINUE
00102   CONTINUE
00104   DO 124 K = 1,KMAX,2
00106     DO 114 J = 1,JMAX,2
00108       DO 112 I = 1,IMAX,2
00110         CRYST3(I,J,K) = 2
00112       CONTINUE
00114     CONTINUE
00116     DO 123 J = 2,JMAX,2
00118       DO 122 I = 2,IMAX,2
00120         CRYST3(I,J,K) = 2
00122       CONTINUE
00123     CONTINUE
00124   CONTINUE
00125   DO 146 K = 3,KMAX,2
00126     DO 134 J = 2,JMAX,2
00128       DO 132 I = 1,IMAX,2
00130         CRYST3(I,J,K) = 0
00132       CONTINUE
00134     CONTINUE
00136     DO 144 J = 1,JMAX,2
00138       DO 142 I = 2,IMAX,2
00140         CRYST3(I,J,K) = 0
00142       CONTINUE
00144     CONTINUE
00146   CONTINUE
00148   K = 1
00150     DO 158 J = 2,JMAX,2
00152       DO 156 I = 1,IMAX,2
00154         CRYST3(I,J,K) = 1
00156       CONTINUE
00158     CONTINUE
00160     DO 172 J = 1,JMAX,2
00162       DO 170 I = 2,IMAX,2
00164         CRYST3(I,J,K) = 1
00170       CONTINUE
00172     CONTINUE
C
C
C
C  END OF THREE DIMENSIONAL CRYSTAL SET UP AND THE BEGINNING OF THE ONE
C  DIMENSIONAL EQUIVELENCE.  THE ONE DIMENSIONAL ARRAY CRYST(L) VALUES
C  ARE IDENTICAL TO THE THREE DIMENSIONAL CRYST3(I,J,K).
C
C
00174   DO 188 K = 1,KMAX
00176     DO 186 J = 1,JMAX
00178       DO 184 I = 1,IMAX
00180         L = I +((J-1)*IMAX) + ((K-1)*PLNMAX)
00182         CRYST(L) =CRYST3(I,J,K)
00184       CONTINUE
00186     CONTINUE
00188   CONTINUE
00210   KMAX1 = KMAX -1
C
C
```

```
C   THE FOLLOWING GROUP OF DO LOOPS DEVELOP THE TWO DIMENSIONAL
C   ARRAY CRNNBR(L,R).  THIS ARRAYS VALUES ARE THE TWELVE NEAREST
C   NEIGHBORS TO THE CRYSTAL SITE L.
C
C
00220      DO 380 K = 2,KMAX1
00230         J = 1
00240           I = 1
00250              L = I +        (K-1)*PLNMAX
00260                CRNNBR(L,1) = IMAX + K*PLNMAX
00270                CRNNBR(L,2) = I + (JMAX-1)*IMAX + K*PLNMAX
00280                CRNNBR(L,3) = I+1 + K*PLNMAX
00290                CRNNBR(L,4) = I + IMAX + K*PLNMAX
00300                CRNNBR(L,5) = IMAX + (JMAX-1)*IMAX + (K-1)*PLNMAX
00310                CRNNBR(L,6) = I+1 + (JMAX-1)*IMAX + (K-1)*PLNMAX
00320                CRNNBR(L,7) = I+1 + IMAX +(K-1)*PLNMAX
00330                CRNNBR(L,8) = IMAX + IMAX + (K-1)*PLNMAX
00340                CRNNBR(L,9) = IMAX + (K-2)*PLNMAX
00350                CRNNBR(L,10) = I + (JMAX-1)*IMAX + (K-2)*PLNMAX
00360                CRNNBR(L,11) = I+1 + (K-2)*PLNMAX
00370                CRNNBR(L,12) = I + IMAX + (K-2)*PLNMAX
00380      CONTINUE
00390      DO 530 K = 2,KMAX1
00400         J = 1
00410            I = IMAX
00420               L = I + (K-1)*PLNMAX
00430                 CRNNBR(L,1) = I-1 + K*PLNMAX
00440                 CRNNBR(L,2) = I + (JMAX-1)*IMAX + K*PLNMAX
00450                 CRNNBR(L,3) = 1 + K*PLNMAX
00460                 CRNNBR(L,4) = I + IMAX +K*PLNMAX
00470                 CRNNBR(L,5) = I-1 + (JMAX-1)*IMAX + (K-1)*PLNMAX
00480                 CRNNBR(L,6) = 1 + (JMAX-1)*IMAX + (K-1)*PLNMAX
00482                 CRNNBR(L,7) = 1 + IMAX + (K-1)*PLNMAX
00484                 CRNNBR(L,8) = I-1 + IMAX + (K-1)*PLNMAX
00490                 CRNNBR(L,9) = I-1 + (K-2)*PLNMAX
00500                 CRNNBR(L,10) = I + (JMAX-1)*IMAX + (K-2)*PLNMAX
00510                 CRNNBR(L,11) = 1 + (K-2)*PLNMAX
00520                 CRNNBR(L,12) = I + IMAX + (K-2)*PLNMAX
00530     CONTINUE
00540      DO 700 K = 2,KMAX1
00550         J = JMAX
00560            I = IMAX
00570              L = K* PLNMAX
00580                 CRNNBR(L,1) = I-1 + (J-1)*IMAX + K*PLNMAX
00590                 CRNNBR(L,2) = I + (J-2)*IMAX + K*PLNMAX
00600                 CRNNBR(L,3) = 1 + (J-1)*IMAX + K*PLNMAX
00610                 CRNNBR(L,4) = I + K*PLNMAX
00620                 CRNNBR(L,5) = I-1 + (J-2)*IMAX + (K-1)*PLNMAX
00630                 CRNNBR(L,6) = 1 + (J-2)*IMAX + (K-1)*PLNMAX
00640                 CRNNBR(L,7) = 1 + (K-1)*PLNMAX
00650                 CRNNBR(L,8) = I-1 + (K-1)*PLNMAX
00660                 CRNNBR(L,9) = I-1 + (J-1)*IMAX + (K-2)*PLNMAX
00670                 CRNNBR(L,10) = I + (J-2)*IMAX + (K-2)*PLNMAX
00680                 CRNNBR(L,11) = 1 + (J-1)*IMAX + (K-2)*PLNMAX
00690                 CRNNBR(L,12) = I +(K-2)*PLNMAX
00700      CONTINUE
00710      DO 850 K = 2,KMAX1
00720         J = JMAX
00730            I = 1
00740              L = I + (J-1)*IMAX + (K-1)*PLNMAX
00750                 CRNNBR(L,1) = IMAX + (J-1)*IMAX + K*PLNMAX
00760                 CRNNBR(L,2) = I + (J-2)*IMAX + K*PLNMAX
00770                 CRNNBR(L,3) = I+1 + (J-1)*IMAX + K*PLNMAX
00780                 CRNNBR(L,4) = I + K*PLNMAX
00790                 CRNNBR(L,5) = IMAX + (J-2)*IMAX + (K-1)*PLNMAX
00800                 CRNNBR(L,6) = I+1 +(J-2)*IMAX + (K-1)*PLNMAX
00810                 CRNNBR(L,7) = I+1 +(K-1)*PLNMAX
00820                 CRNNBR(L,8) = IMAX + (K-1)*PLNMAX
```

```
              CRNNBR(L,9) = IMAX + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,10) = I + (J-2)*  IMAX + (K-2)*PLNMAX
              CRNNBR(L,11) = I+1 + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,12) = I + (K-2)*PLNMAX
   CONTINUE
JMAX1 = JMAX -1
IMAX1 = IMAX -1
   DO 1050 K = 2,KMAX1
      I = 1
        DO 1040 J = 2,JMAX1
           L = I + (J-1)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,1) = IMAX + (J-1)*IMAX + (K*PLNMAX)
              CRNNBR(L,2) = I + (J-2)*IMAX + K*PLNMAX
              CRNNBR(L,3) = I+1 + (J-1)*IMAX + K*PLNMAX
              CRNNBR(L,4) = I + J*IMAX + K*PLNMAX
              CRNNBR(L,5) = IMAX + (J-2)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,6) = I+1 + (J-2)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,7) = I+1 +(J*IMAX) + (K-1)*PLNMAX
              CRNNBR(L,8) = IMAX + J*IMAX + (K-1)*PLNMAX
              CRNNBR(L,9) = IMAX + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,10) = I + (J-2)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,11) = I+1 + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,12) = I +   J*IMAX + (K-2)*PLNMAX
      CONTINUE
   CONTINUE
   DO 1230 K = 2,KMAX1
      J = 1
        DO 1220 I = 2,IMAX1
           L = I + (K-1)*PLNMAX
              CRNNBR(L,1) = I-1 + K*PLNMAX
              CRNNBR(L,2) = I + (JMAX-1)*IMAX + K*PLNMAX
              CRNNBR(L,3) = I+1 + K*PLNMAX
              CRNNBR(L,4) = I + IMAX +K*PLNMAX
              CRNNBR(L,5) = I-1 + (JMAX-1)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,6) = I+1 + (JMAX-1)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,7) = I+1 + IMAX + (K-1)*PLNMAX
              CRNNBR(L,8) = I-1 + IMAX + (K-1)*PLNMAX
              CRNNBR(L,9) = I-1 + (K-2)*PLNMAX
              CRNNBR(L,10) = I + (JMAX-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,11) = I+1 + (K-2)*PLNMAX
              CRNNBR(L,12) = I + IMAX + (K-2)*PLNMAX
        CONTINUE
   CONTINUE
   OO 1410 K = 2,KMAX1
      I = IMAX
        OO 1400 J = 2,JMAX1
           L = I + (J-1)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,1) = I-1 + (J-1)*IMAX + K*PLNMAX
              CRNNBR(L,2) = I + (J-2)*IMAX + (K*PLNMAX)
              CRNNBR(L,3) = 1 + (J-1)*IMAX + K*PLNMAX
              CRNNBR(L,4) = I +(J*IMAX) + K*PLNMAX
              CRNNBR(L,5) = I-1 + (J-2)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,6) = 1 + (J-2)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,7) = 1 + J*IMAX + (K-1)*PLNMAX
              CRNNBR(L,8) = I-1 +(J*IMAX) + (K-1)*PLNMAX
              CRNNBR(L,9) = I-1 + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,10) = I + (J-2)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,11) = 1 + (J-1)*IMAX + (K-2)*PLNMAX
              CRNNBR(L,12) = I + J*IMAX + (K-2)*PLNMAX
          CONTINUE
   CONTINUE
   DO 1590 K = 2, KMAX1
      J = JMAX
        DO 1580 I = 2,IMAX1
           L = I + (J-1)*IMAX + (K-1)*PLNMAX
              CRNNBR(L,1) = I-1 + (J-1)*IMAX +K*PLNMAX
              CRNNBR(L,2) = I + (J-2)*IMAX + K*PLNMAX
              CRNNBR(L,3) = I+1 + (J-1)*IMAX + K*PLNMAX
```

```
01490                 CRNNBR(L,4) = I + K*PLNMAX
01500                 CRNNBR(L,5) = I-1 + (J-2)*IMAX + (K-1)*PLNMAX
01510                 CRNNBR(L,6) = I+1 + (J-2)*IMAX + (K-1)*PLNMAX
01520                 CRNNBR(L,7) = I+1 + (K-1)*PLNMAX
01530                 CRNNBR(L,8) = I-1 + (K-1)*PLNMAX
01540                 CRNNBR(L,9) = I-1 + (J-1)*IMAX + (K-2)*PLNMAX
01550                 CRNNBR(L,10) = I + (J-2)*IMAX + (K-2)*PLNMAX
01560                 CRNNBR(L,11) = I+1 + (J-1)*IMAX + (K-2)*PLNMAX
01570                 CRNNBR(L,12) = I + (K-2)*PLNMAX
01580           CONTINUE
01590     CONTINUE
01600     DO 1780 K = 2,KMAX1
01610        DO 1770 J = 2,JMAX1
01620           DO 1760 I = 2,IMAX1
01630             L = I + (J-1)*IMAX + (K-1)*PLNMAX
01640                 CRNNBR(L,1) = I-1 + (J-1)*IMAX + K*PLNMAX
01650                 CRNNBR(L,2) = I + (J-2)*IMAX + K*PLNMAX
01660                 CRNNBR(L,3) = I+1 + (J-1)*IMAX + K*PLNMAX
01670                 CRNNBR(L,4) = I + J*IMAX + K*PLNMAX
01680                 CRNNBR(L,5) = I-1 + (J-2)*IMAX + (K-1)*PLNMAX
01690                 CRNNBR(L,6) = I+1 + (J-2)*IMAX + (K-1)*PLNMAX
01700                 CRNNBR(L,7) = I+1 + J*IMAX + (K-1)*PLNMAX
01710                 CRNNBR(L,8) = I-1 + J*IMAX + (K-1)*PLNMAX
01720                 CRNNBR(L,9) = I-1 + (J-1)*IMAX + (K-2)*PLNMAX
01730                 CRNNBR(L,10) = I + (J-2)*IMAX + (K-2)*PLNMAX
01740                 CRNNBR(L,11) = I+1 + (J-1)*IMAX + (K-2)*PLNMAX
01750                 CRNNBR(L,12) = I + J*IMAX + (K-2)*PLNMAX
01760           CONTINUE
01770        CONTINUE
01780     CONTINUE
01790     DO 1960 J = 2,JMAX1
01800        K = 1
01810           DO 1950 I = 2,IMAX1
01820             L = I + (J-1)*IMAX
01830                 CRNNBR(L,1) = I-1 + (J-1)*IMAX + PLNMAX
01840                 CRNNBR(L,2) = L
01850                 CRNNBR(L,3) = L
01860                 CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
01870                 CRNNBR(L,5) = L
01880                 CRNNBR(L,6) = L
01890                 CRNNBR(L,7) = I+1+(J-1)*IMAX + PLNMAX
01900                 CRNNBR(L,8) = L
01910                 CRNNBR(L,9) = L
01920                 CRNNBR(L,10) = I + J*IMAX + PLNMAX
01930                 CRNNBR(L,11) = L
01940                 CRNNBR(L,12) = L
01950          CONTINUE
01960     CONTINUE
01970     DO 2130 I = 2,IMAX1
01980        K = 1
01990          J = 1
02000             L = I
02010                 CRNNBR(L,1) = I-1 + PLNMAX
02020                 CRNNBR(L,2) = L
02030                 CRNNBR(L,3) = L
02040                 CRNNBR(L,4) = I + (JMAX-1)*IMAX + PLNMAX
02050                 CRNNBR(L,5) = L
02060                 CRNNBR(L,6) = L
02070                 CRNNBR(L,7) = I+1 +PLNMAX
02080                 CRNNBR(L,8) = L
02090                 CRNNBR(L,9) = L
02100                 CRNNBR(L,10) = I + IMAX + PLNMAX
02110                 CRNNBR(L,11) = L
02120                 CRNNBR(L,12) = L
02130     CONTINUE
02140     DO 2300 J = 2,JMAX1
02150        K = 1
02160           I = IMAX
```

```
           L = I + (J-1)*IMAX
              CRNNBR(L,1) = I-1 + (J-1)*IMAX +PLNMAX
              CRNNBR(L,2) = L
              CRNNBR(L,3) = L
              CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
              CRNNBR(L,5) = L
              CRNNBR(L,6) = L
              CRNNBR(L,7) =   1 + (J-1)*IMAX + PLNMAX
              CRNNBR(L,8) = L
              CRNNBR(L,9) = L
              CRNNBR(L,10) = I + J*IMAX + PLNMAX
              CRNNBR(L,11) = L
              CRNNBR(L,12) = L
   CONTINUE
   DO 2470 I = 2,IMAX1
      K = 1
         J = JMAX
           L = I + (J-1)*IMAX
              CRNNBR(L,1) = I-1 + (J-1)*IMAX + PLNMAX
              CRNNBR(L,2) = L
              CRNNBR(L,3) = L
              CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
              CRNNBR(L,5) = L
              CRNNBR(L,6) = L
              CRNNBR(L,7) = I+1 + (J-1)*IMAX + PLNMAX
              CRNNBR(L,8) = L
              CRNNBR(L,9) = L
              CRNNBR(L,10) = I + PLNMAX
              CRNNBR(L,11) = L
              CRNNBR(L,12) = L
   CONTINUE
   DO 2640 J = 2,JMAX1
      K = 1
        I = 1
           L = I + (J-1)*IMAX
              CRNNBR(L,1)=IMAX + (J-1)*IMAX + PLNMAX
              CRNNBR(L,2) = L
              CRNNBR(L,3) = L
              CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
              CRNNBR(L,5) = L
              CRNNBR(L,6) = L
              CRNNBR(L,7) = I+1 + (J-1)*IMAX + PLNMAX
              CRNNBR(L,8) = L
              CRNNBR(L,9) = L
              CRNNBR(L,10) = I + J*IMAX + PLNMAX
              CRNNBR(L,11) = L
              CRNNBR(L,12) = L
  CONTINUE
   I = 1
   J = 1
   K = 1
              CRNNBR(1,1) = IMAX + PLNMAX
              CRNNBR(1,2) = 1
              CRNNBR(1,3) = 1
              CRNNBR(1,4) = I + (JMAX-1)*IMAX + PLNMAX
              CRNNBR(1,5) = 1
              CRNNBR(1,6) = 1
              CRNNBR(1,7) = I+1 + PLNMAX
              CRNNBR(1,8) = 1
              CRNNBR(1,9) = 1
              CRNNBR(1,10) = I + IMAX + PLNMAX
              CRNNBR(1,11) = 1
              CRNNBR(1,12) = 1
   I = IMAX
   L = IMAX
              CRNNBR(L,1) = I-1 + PLNMAX
              CRNNBR(L,2) = L
              CRNNBR(L,3) = L
```

```
02830                CRNNBR(L,4) = I + (JMAX-1)*IMAX + PLNMAX
02840                CRNNBR(L,5) = L
02850                CRNNBR(L,6) = L
02860                CRNNBR(L,7) = 1 + PLNMAX
02870                CRNNBR(L,8) = L
02880                CRNNBR(L,9) = L
02890                CRNNBR(L,10) = I + IMAX + PLNMAX
02900                CRNNBR(L,11) = L
02910                CRNNBR(L,12) = L
02920     I = IMAX
02922     J = JMAX
02924     K = 1
02930            L = PLNMAX
02940                CRNNBR(L,1) = I-1 + (J-1)*IMAX + PLNMAX
02950                CRNNBR(L,2) = L
02960                CRNNBR(L,3) = L
02970                CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
02980                CRNNBR(L,5) = L
02990                CRNNBR(L,6) = L
03000                CRNNBR(L,7) =   1 +(J-1)*IMAX + PLNMAX
03010                CRNNBR(L,8) = L
03020                CRNNBR(L,9) = L
03030                CRNNBR(L,10) = I + PLNMAX
03040                CRNNBR(L,11) = L
03050                CRNNBR(L,12) = L
03060     I = 1
03070        L = 1 + (J-1)*IMAX
03080                CRNNBR(L,1) = IMAX + (J-1)*IMAX + PLNMAX
03090                CRNNBR(L,2) = L
03100                CRNNBR(L,3) = L
03110                CRNNBR(L,4) = I + (J-2)*IMAX + PLNMAX
03120                CRNNBR(L,5) = L
03130                CRNNBR(L,6) = L
03140                CRNNBR(L,7)=I+1 + (J-1)*IMAX + PLNMAX
03150                CRNNBR(L,8) = L
03160                CRNNBR(L,9) = L
03170                CRNNBR(L,10) = I + PLNMAX
03180                CRNNBR(L,11) = L
03190                CRNNBR(L,12) = L
C
C
C   THE FOLLOWING CODE IS USED TO SEND THE DEVELOPED ARRAYS TO
C   AN OUTPUT FILE.
C
C
                     PRINT 3193, 'THIS IS THE LATICE SITES VS PLANES'
03193                FORMAT ('1',47X,A34)
                       PRINT 3195, 'SITE #', 'PLANE'
03195                  FORMAT ('-',10X,A8,3X,A5)
                     DO 3194 L = 1,LMAX
                     PRINT 3196, L, ZPLN(L)
03194                CONTINUE
03196                FORMAT (' ',10X,I8,5X,I3)
03200     PRINT 3210, ' NEAREST NEIGHBOR TABLE '
03210     FORMAT ('1',53X,A24)
03220       PRINT 3230,'L',1,2,3,4,5,6,7,8,9,10,11,12
03230       FORMAT ('-',4X,A1,9X,I1,7X,I1,7X,I1,7X,I1,7X,I1,7X,I1,7X,I1,
     1     7X,I1,7X,I1,6X,I2,6X,I2,6X,I2)
03240         LMAX1 = LMAX - PLNMAX
03250            DO 3270 L = 1,LMAX1
03260              PRINT 3280, L,(CRNNBR(L,R), R=1,12)
03270             CONTINUE
03280               FORMAT (' ',13I8)
                  DO 3287 K = 1,KMAX
                      PRINT 3284, 'THIS IS PLANE NUMBER', K
                        DO 3282 J = 1,JMAX
                           PRINT 3286, (CRYST3(I,J,K), I = 1,IMAX)
03282                   CONTINUE
```

```
03284                FORMAT ('1',5X,A20,3X,I3)
03286                      FORMAT (' ',4I4)
03287          CONTINUE
03288 WRITE (1,3289)IMAX,IMAX1,JMAX,JMAX1,KMAX,KMAX1,PLNMAX,LMAX,LMAX1,
     1              ISEED1,ISEED2,MTMSTP,T1,AVGTM,PLSTM,OUTPUT
03289 FORMAT(' ',9I8,' ',2I16,' ',5I8)
         DO 3290 L = 1,LMAX
          WRITE (1,3291)CRYST(L)
03290    CONTINUE
03291    FORMAT (' ',I8)
         DO 3300 L = 1,LMAX1
         DO 3292 R = 1,12
           WRITE (1,3301)             CRNNBR(L,R)
03292   CONTINUE
03300    CONTINUE
03301      FORMAT (' ',3I8)
         DO 3310 L = 1,LMAX
           WRITE (1,3311)ZPLN(L)
03310    CONTINUE
03311       FORMAT (' ',I8)
03320 STOP
      END
```

APPENDIX C

'DIFFUSE' CODE LISTING

The following is the computer code for the diffusion simulation named 'DIFFUSE'.

```
C                             'DIFFUSE'
C                         MARK R. POLNASZEK
C                             JUNE 1986
C
C
C
C   THE CODE 'DIFFUSE' HAS BEEN DEVELOPED TO SIMULATE THE DIFFUSION OF
C   AN ATOM INTERSTITIALLY THROUGH A FACE-CENTERED-CUBIC CRYSTAL LATTICE
C   THIS LATTICE IS AN INPUT ARRAY THAT WAS DEVELOPED IN ANOTHER PROGRAM
C   AND INPUTED ALONG WITH EACH SITES NEAREST-NEIGHBOR, AND EACH SITES
C   PLANAR POSITION.   THE METHOD THAT 'DIFFUSE' USES TO MOVE THE
C   DIFFUSING ATOMS THROUGH THE CRYSTAL ARRAY IS BY A MONTE CARLO METHOD
C   THE SIMULATION THEN DEVELOPS THE DIFFUSION COEFFICIENT BY FICK'S
C   FIRST AND SECOND LAW.
C
C
C
C   THE FOLLOWING DEFINES THE VARIABLES AS REAL OR INTEGER AND
C   DIMENSIONS THE VARIABLE ARRAYS' MEMORY ALLOCATION.
C
C
C
      INTEGER PLNMAX, CRYST(1300),CRYST3(08,08,16),CRNNBR(1300,12),R,
     1 MAVGTM, ZPLN(1300),AVGTM,TMSTP2,  CONS(1100,16),T1,MTMSTP,TMSTP,
     2 AVGTM1,AVGTM2,AT1,AVGTM3,ATMSTP,OUTPUT
      REAL*4  RDM1(1300),RDM2(1300),ACONS(210,16), AFLUX(210,16) ,
     1 ADERC1(210,16) ,ADERC2(210,16) ,ADERT(210,16) ,DIFCO1(210,16) ,
     2 DIFCO2(210,16), FLUX(1100,16), PLNMX1,CVOL(1100,16),
     3 DERC1(1100,16),DERC2(1100,16),DERT(1100,16), ACVOL(210,16),
     4 DERDC(210,16),DERT2(210,16)
      READ(5,*) IMAX,IMAX1,JMAX,JMAX1,KMAX,KMAX1,PLNMAX,LMAX,LMAX1,
     1          ISEED1,ISEED2,MTMSTP,T1,AVGTM,TMSTP2,OUTPUT
        DO    10  L = 1,LMAX
C
C
C   THE FOLLOWING CODE READS IN THE REQUIRED INPUT AND ECHOS
C   THE INPUT INTO AN OUTPUT FILE FOR INSPECTON.
C
C
              READ (5,*) CRYST(L)
00010   CONTINUE
        DO    20  L = 1,LMAX1
          DO    15  R = 1,12
                READ (5,*) CRNNBR(L,R)
00015     CONTINUE
00020   CONTINUE
        DO    30  L = 1,LMAX
              READ (5,*) ZPLN(L)
00030   CONTINUE
          IF (MTMSTP .EQ. 0)      GOTO 55
        DO    40  TMSTP =  1,MTMSTP
          DO    35  K = 1,KMAX1
               READ (5,*) FLUX(TMSTP,K)
00035     CONTINUE
00040   CONTINUE
        DO    50  TMSTP =  1,MTMSTP
          DO    45  K = 1,KMAX1
               READ (5,*) CONS(TMSTP,K)
00045     CONTINUE
00050   CONTINUE
00055   IF (OUTPUT .NE. 1) GOTO 300
00056     WRITE (6,60) 'INPUTED VALUES'
00060     FORMAT ('1',33X,A14)
            WRITE (6,70) 'IMAX','IMAX1','JMAX','JMAX1','KMAX','KMAX1',
     1                   'PLNMAX','LMAX','LMAX1','TIME AVG','TMSTP2'
00070       FORMAT ('-',11A10)
          WRITE (6,80) IMAX,IMAX1,JMAX,JMAX1,KMAX,KMAX1,PLNMAX,LMAX,
     1                 LMAX1,AVGTM,TMSTP2
```

```
00080       FORMAT (' ',11I10)
              WRITE (6,90) 'RANDOM GENERATOR SEEDS'
00090         FORMAT ('-',29X,A22)
            WRITE (6,0100) ISEED1,ISEED2
00100       FORMAT (' ',10X,2I16)
              WRITE (6,0110) 'TIME CONSTANTS'
00110         FORMAT ('-',33X,A14)
            WRITE (6,0120) 'START TIME', 'END TIME'
00120       FORMAT ('-',10X,2A10)
              WRITE (6,0130) T1,MTMSTP
00130         FORMAT (' ',10X,2I10)
            WRITE (6,0140)  'INPUTED CRYSTAL'
00140       FORMAT ('1',32X,A15)
            WRITE (6,145) 'LATICE ATOM - 2','DIFFUSING ATOM - 1',
     1                        'VACANCY - 0'
00145       FORMAT ('-',3A25)
              WRITE (6,146) 'SITE', 'ATOM TYPE'
00146         FORMAT ('-',10X,2A10)
         DO    150  L = 1,LMAX
               WRITE (6,160) L, CRYST(L)
00150    CONTINUE
00160          FORMAT (' ',10X,2I10)
             WRITE (6,170) 'NEAREST NEIGHBOR TABLE'
00170        FORMAT ('1',29X,A22)
              WRITE (6,0175) 'L',1,2,3,4,5,6,7,8,9,10,11,12
00175         FORMAT ('-',A8,12I8)
         DO   180  L = 1,LMAX1
               WRITE (6,190) L,(CRNNBR(L,R), R=1,12)
00180    CONTINUE
00190          FORMAT  (' ',13I8)
C
C
C   THE CODE BELOW SETS UP CONSTANTS THAT ARE USED REPEATEDLY
C   THROUGHOUT THE REMAINDER OF THE PROGRAM.
C
C
00300     PLNMX1  = 2/FLOAT(PLNMAX)
          T1 = T1 + MTMSTP
          MTMSTP = MTMSTP + TMSTP2
          AVGTM1 = AVGTM - 99
          AT1 = INT(FLOAT(T1)/AVGTM)+1
          MAVGTM = INT(FLOAT(MTMSTP)/AVGTM)
C
C
C   THE BELOW CODE IS THE BEGINNING OF THE MAJOR DO LOOP.  THIS
C   DO LOOP COMPLETES ALL THE REQUIRED TIMESTEPS FOR THE PROGRAM.
C   ALL CALCULATIONS OF THE SIMULATION ARE COMPLETED WITHIN THIS
C   DO LOOP.
C
C
        DO 1050 ATMSTP = AT1,MAVGTM
C
C
C   THE FOLLOWING DO LOOP COUNTS ALL DIFFUSING ATOMS IN EACH
C   PLANE.  THIS IS THE CONCENTRATION OF EACH PLANE.  THE DO LOOP
C   ALSO CALCULATES THE CONCENTRATION PER UNIT VOLUME OF EACH
C   CRYSTAL PLANE.
C
C
         DO 1001  TMSTP = 1,AVGTM
               DO   301  L = 1,LMAX
                K = ZPLN(L)
                  IF (CRYST(L) .EQ. 1)  THEN
                  CONS(TMSTP,K) =CONS(TMSTP,K) + 1
                  CVOL(TMSTP,K) = CVOL(TMSTP,K) + PLNMX1
                  END IF
00301          CONTINUE
C
```

```
C
C   THE FOLLOWING CONSTANT IS DETERMINED FOR USE IN THE OUTPUT FILE.
C
C
              AVGTM2 = (ATMSTP*AVGTM)-(100 -TMSTP)
C
C
C   THE FOLLOWING IF STATEMENT SHORTENS THE NUMBER OF SITES THAT THE
C   SIMULATION WILL LOOK AT IF THE STATEMENT IS TRUE, AND ASSIGNS A
C   CONSTANT NUMBER OF SITES IF THE STATEMENT IS FALSE.
C
C
              IF (AVGTM2.LE. KMAX1) THEN
              INMBR = AVGTM2* PLNMAX
              GOTO 310
              END IF
              INMBR = LMAX1
C
C
C   THE FOLLOWING SUB-ROUTINE CALLED FROM THE IMSL LIBRARY SETS UP
C   TWO ARRAYS OF UNIFORM RANDOM NUMBERS.
C
C
00310           CALL SRND(ISEED1,RDM1,INMBR,2,0)
00311           CALL SRND(ISEED2,RDM2,INMBR,2,0)
C
C
C   THE DO 999 DO LOOP LOOKS AT EACH SITE IN THE CRYSTAL ARRAY
C   TO DETERMINE IF A DIFFUSING ATOM IS IN THAT SITE.  IF THERE
C   IS A DIFFUSING ATOM IN THAT SITE THE CODE MOVES IT TO NEW
C   SITE IF IT CAN.
C
C
00312      DO   999  LRDM = 1,INMBR
00313         L = INT(INMBR*RDM1(LRDM)) + 1
00314         R = INT (12 * RDM2(LRDM)) + 1
              L1 = CRNNBR(L,R)
C
C
C   THE FOLLOWING IF STATEMENT DETERMINES IF THE SITE IS OCCUPIED
C   BY A DIFFUSING ATOM.  IF THE STATEMENT IS FALSE THE CODE ALLOWS
C   THE DO LOOP TO CONTINUE.
C
C
         IF (CRYST(L) .NE. 1) GOTO 999
C
C
C   THE FOLLOWING DO LOOP IS USED TO SEND FIRST PLANE ATOMS TO
C   THEIR SPECIAL CODE.
C
C
00316            IF (ZPLN(L) .EQ. 1)  GOTO 320
C
C
C   THE NEXT DO LOOP MOVES THE ATOM TO ITS NEW SITE .
C
C
00317    IF (CRYST(L1) .EQ. 0) GOTO (400,400,400,400,500,500,500,
     1                                  500,600,600,600,600) R
00318    GOTO 999
C
C
C   THE FOLLOWING IF STATEMENT MOVES THE FIRST PLANE ATOMS.
C
C
00320    IF (CRYST(L1) .EQ. 0)  THEN
          FLUX(TMSTP,1) =FLUX(TMSTP,1) + PLNMX1
          CRYST(L1) = CRYST(L)
          END IF
         GOTO 999
C
C
C   THE FOLLOWING CODE MOVES ALL ATOMS OTHER THAN FIRST PLANE
C   IF THE NEAREST-NEIGHBOR SELECTED WAS IN THE FORWARD PLANE.
C
C
00400         K = ZPLN(L)
00401         FLUX(TMSTP,K) = FLUX(TMSTP,K) + PLNMX1
00403         IF (ZPLN(L1) .EQ. KMAX)  GOTO 410
00404         CRYST(L1) = CRYST(L)
00410         CRYST(L) = 0
              GOTO 999
C
C
C    THE FOLLOWING CODE MOVES the atom if it is to stay in the same
C    PLANE.
C
C
00500              CRYST(L1) = CRYST(L)
                   CRYST(L) = 0
                   GOTO 999
C
C
C    THE FOLLOWING CODE IS USED IF THE ATOM IS TO MOVE INTO THE
C    PLANE BEHIND THE DIFFUSING ATOMS INITIAL POSITION.
C
C
00600         K = ZPLN(L1)
              FLUX(TMSTP,K) = FLUX(TMSTP,K) - PLNMX1
              CRYST(L1) = CRYST(L)
              CRYST(L) = 0
00999   CONTINUE
01001    CONTINUE
C
C
C     THE FOLLOWING DO LOOP FINDS THE CONCENTRATION FOR THE END OF THE
C     LAST TIMESTEP.
C
C
           DO 1002 L = 1,LMAX
            K = ZPLN(L)
            IF (CRYST(L) .EQ. 1) THEN
            CONS(AVGTM+1,K) = CONS(AVGTM+1,K) + 1
            CVOL(AVGTM+1,K) = CVOL(AVGTM+1,K) + PLNMX1
            END IF
01002      CONTINUE
C
C
C    THE FOLLOWING DO LOOPS FIGURE THE DERIVATIVES NEEDED TO
C    DETERMINE THE DIFFUSION COEFFICIENTS.
C
C
        DO 1004 TMSTP = 1,AVGTM
           DO 1003 K = 2,KMAX1
              DERC1(TMSTP,K) =(CVOL(TMSTP,K+1) - CVOL(TMSTP,K-1))/2
01003      CONTINUE
01004    CONTINUE
         DO 1006 TMSTP = 1,AVGTM
            DO 1005 K = 3,KMAX1-1
       DERC2(TMSTP,K)=(DERC1(TMSTP,K+1)-DERC1(TMSTP,K-1))/2
01005       CONTINUE
01006    CONTINUE
         DO  1008 TMSTP = 1,AVGTM
             DO 1007 K = 1,KMAX1
                DERT(TMSTP  ,K) = CVOL(TMSTP+1,K) - CVOL(TMSTP,K)
01007        CONTINUE
```

```
01008    CONTINUE
C
C
C   THE FOLLOWING DO LOOP AVERAGES THE DETERMINED VALUES AS NEEDED
C   TO DETERMINE THE DIFFUSION COEFFICIENT.
C
C
         DO 1020 TMSTP = 1,AVGTM
               DO 1010 K = 1,KMAX1
        ACONS(ATMSTP,K) = ACONS(ATMSTP,K) + FLOAT(CONS(TMSTP,K))/AVGTM
        AFLUX(ATMSTP,K) = AFLUX(ATMSTP,K) +       FLUX(TMSTP,K) /AVGTM
        ADERC1(ATMSTP,K) = ADERC1(ATMSTP,K) +      DERC1(TMSTP,K) /AVGTM
        ADERC2(ATMSTP,K) = ADERC2(ATMSTP,K) +      DERC2(TMSTP,K) /AVGTM
        ADERT(ATMSTP,K) = ADERT(ATMSTP,K) +       DERT(TMSTP,K) /AVGTM
        ACVOL(ATMSTP,K) = ACVOL(ATMSTP,K) + CVOL(TMSTP,K)/AVGTM
01010         CONTINUE
01020    CONTINUE
C
C
C    THE FOLLOWING DO LOOP DETERMINES THE DIFFUSION COEFFICIENT
C    BY BOTH THE FIRST LAW (DIFCO1) AND SECOND LAW (DIFCO2) METHOD.
C
C
            DO 1030 K = 3,KMAX1-1
               DIFCO1(ATMSTP,K) = -1* AFLUX(ATMSTP,K)/ADERC1(ATMSTP,K)
               DIFCO2(ATMSTP,K) = ADERT(ATMSTP,K)/ADERC2(ATMSTP,K)
01030       CONTINUE
C
C
C   THE FOLLOWING DO LOOP DETERMINES THE FIRST DERIVATIVE OF THE
C   DIFFUSION COEFFICIENT AND BY FICK'S SECOND LAW (NON-LINEAR)
C   THE TIME DERIVATIVE OF THE CONCENTRATION.
C
C
       DO 1031 K = 3,KMAX1-2
          DERDC(ATMSTP,K) = DIFCO1(ATMSTP,K+1) - DIFCO1(ATMSTP,K-1)
          DERT2(ATMSTP,K) = DERDC(ATMSTP,K)*ADERC1(ATMSTP,K) +
     1                      DIFCO1(ATMSTP,K)*ADERC2(ATMSTP,K)
01031    CONTINUE
C
C
C   THE FOLLOWING DO LOOP CLEARS THE CONTENTS OF THE NAMED ARRAYS.
C
C
         IF (ATMSTP .EQ. MAVGTM) GOTO 1050
        DO 1033 TMSTP = 1,AVGTM+1
           DO 1032 K = 1,KMAX
                CONS(TMSTP,K) = 0
                FLUX(TMSTP,K) = 0
                DERC1(TMSTP,K) = 0
                DERC2(TMSTP,K) = 0
                DERT(TMSTP,K)  = 0
                CVOL(TMSTP,K) = 0
01032   CONTINUE
01033   CONTINUE
01050   CONTINUE
C
C
C   THE FOLLOWING CODE WRITES THE GENERATED VALUES OF THE SIMULATION
C   TO AN OUTPUT FILE FOR ANALYSIS.
C
C
        GOTO (1051,1191,1241),OUTPUT
01051       WRITE (6,1060) 'OUTPUTED VALUES'
01060       FORMAT ('1',33X,A15)
              WRITE(6,1070)'IMAX','IMAX1','JMAX','JMAX1','KMAX','KMAX1',
     1                      'PLNMAX','LMAX','LMAX1'
01070         FORMAT ('-',9A8)
```

```
            WRITE(6,1080)IMAX,IMAX1,JMAX,JMAX1,KMAX,KMAX1,PLNMAX,LMAX,
     1                    LMAX1
01080       FORMAT (' ',9I8)
              WRITE(6,1090)'RANDOM GENERATOR SEEDS'
01090         FORMAT ('-',29X,A22)
            WRITE (6,1100) ISEED1,ISEED2
01100       FORMAT (' ',10X,2I16)
              WRITE (6,1110) 'TIME CONSTANTS'
01110         FORMAT ('-',33X,A14)
            WRITE (6,1120) 'START TIME', 'END TIME'
01120       FORMAT ('-',10X,2A10)
              WRITE (6,1130) T1,MTMSTP
01130         FORMAT (' ',10X,2I10)
            WRITE (6,1140)  'INPUTED CRYSTAL'
01140       FORMAT ('1',32X,A15)
            WRITE(6,1145) 'LATICE ATOM - 2','DIFFUSING ATOM - 1',
     1                     'VACANCY - 0'
01145       FORMAT ('-',3A25)
              WRITE(6,1146) 'SITE', 'ATOM TYPE'
01146         FORMAT ('-',10X,2A10)
         DO   1150  L = 1,LMAX
               WRITE(6,1160) L, CRYST(L)
01150    CONTINUE
01160          FORMAT (' ',10X,2I10)
             WRITE(6,1170) 'NEAREST NEIGHBOR TABLE'
01170        FORMAT ('1',29X,A22)
              WRITE (6,1175) 'L',1,2,3,4,5,6,7,8,9,10,11,12
01175         FORMAT ('-',A8,12I8)
         DO  1180  L = 1,LMAX1
               WRITE(6,1190) L,(CRNNBR(L,R), R=1,12)
01180    CONTINUE
01190          FORMAT  (' ',13I8)
01191    DO  1210  TMSTP = AVGTM1,AVGTM
           AVGTM3 = (MAVGTM*AVGTM) - (100-TMSTP)
           WRITE (6,1220)  'PARAMETERS FOR TIMESTEP #', AVGTM3
           WRITE  (6,1230) 'PLANE', 'CON  T-','C/VOL T-','FLUX',
     1           '1ST OERV','2ND DERV','TIME DERV', 'CON T+','C/VOL T+'
           DO  1200  K = 1,KMAX1
                WRITE  (6,1240)  K,CONS(TMSTP,K),CVOL(TMSTP,K),
     1      FLUX(TMSTP,K),  DERC1(TMSTP,K),DERC2(TMSTP,K),DERT(TMSTP,K),
     2                              CONS(TMSTP+1,K), CVOL(TMSTP+1,K)
01200      CONTINUE
01210    CONTINUE
01220      FORMAT ('1',A23,I6)
01230      FORMAT ('-',9A10)
01240      FORMAT (' ',2I10,5F10.5,I10,F10.5)
01241    DO 1350 ATMSTP = 1,MAVGTM
            WRITE(6,1360) 'AVERAGE PARAMETERS FOR TIMESTEPS',
     1     ((ATMSTP-1)*AVGTM)+1,'TO',ATMSTP*AVGTM
         WRITE (6,1370) 'PLANE','AVG CONC','AVG C/VOL','AVG FLUX',
     1       '1ST DER',     '2ND DER',     'TIME DER','DIF COEF 1',
     2   'DIF COEF 2', 'DIF CO DER', 'TIME DER ?'
            DO 1340 K = 1,KMAX1
         WRITE (6,1380) K,ACONS(ATMSTP,K),ACVOL(ATMSTP,K),
     1   AFLUX (ATMSTP,K),ADERC1(ATMSTP,K),ADERC2(ATMSTP,K),
     2   ADERT (ATMSTP,K),OIFCO1(ATMSTP,K),OIFCO2(ATMSTP,K),
     3   DERDC(ATMSTP,K),OERT2(ATMSTP,K)
01340      CONTINUE
01350    CONTINUE
01360  FORMAT ('1',A31,I5,A2,I5)
01370  FORMAT ('-',11A11)
01380  FORMAT (' ',I11,10F11.5)
         STOP
         END
```

APPENDIX D

SAMPLE INPUT FOR 'DIFSET'

The following is the items that need to be inputed into the code 'DIFSET'. The method to input this data is to put it in the same order as shown without the titles. There is no format to the length of each data, but insure that there are two blanks between each input.

```
  IMAX      JMAX    KMAX    MTMSTP    T1      ISEED1          ISEED2
   04        04      06       00      01    447586930        88475732
 AVGTM     OUTPUT
  100        1
```

APPENDIX E

SAMPLE INPUT FOR `DIFFUSE'

The following input for the code `DIFFUSE' is created from the output file, FILE FT01F001, from the code `DIFSET'. This is accomplished by renaming the file as INPOL2 DATA. The titles seen in the sample are not a part of the actual input.

Sample input of the constants required for 'diffuse'.
The sample input contains titles that would not be in the
actual input data.

```
IMAX     IMAX1    JMAX     JMAX1    KMAX     KMAX1   PLNMAX   LMAX     LMAX1
  4        3        4        3        6        5       16       96       80



  RANDOM NUMBER GENERATOR SEEDS
   447586930
    88475732



  MTMSTP     T1     AVGTM    TMSTP2     OUTPUT
     0        1      100      100         1
```

Sample input for the CRYST(L) array.

```
      2
      1
      2
      1
      1
      2
      1
      2
      2
      1
      2
      0
      2
      0
      0
      2
      0
      2
      2
      0
      .
      .
      .
      2
      0
      2
      0
      2
      2
      0
      2
      0
      0
      2
      0
      2
      2
      0
      2
      0
```

Sample input for the CRNNBR(L,R) array.

```
20
 1
 1
29
 1
 1
18
 1
 1
21
 1
 1
17
 2
 2
30
 2
 2
 .
 .
 .
63
60
61
52
```

Sample input for the ZPLN(L) array.

```
1
1
.
.
.
1
2
2
.
.
.
2
.
.
.
6
6
.
.
.
6
```

APPENDIX F

SAMPLE OUTPUT OF `DIFSET'

The sample output shown in this Appendix is the some of the output used to check the code for accuracy. This is not all of the output generated, but a representation of what is generated.

```
                              THIS IS THE LATICE SITES VS PLANES


SITE #   PLANE
     1       1
     2       1
     3       1
     4       1
     5       1
     6       1
     7       1
     8       1
     9       1
    10       1
    11       1
    12       1
    13       1
    14       1
    15       1
    16       1
    17       2
    18       2
    19       2
    20       2
    21       2
    22       2
    23       2
    24       2
    25       2
    26       2
    27       2
    28       2
    29       2
    30       2
    31       2
    32       2
    33       3
    34       3
    35       3
    36       3
    37       3
    38       3
    39       3
    40       3
    41       3
    42       3
    43       3
    44       3
    45       3
    46       3
    47       3
    48       3
    49       4
    50       4
    51       4
    52       4
    53       4
    54       4
    55       4
    56       4
    57       4
    58       4
    59       4
    60       4
    61       4
    62       4
    63       4
    64       4
```

NEAREST NEIGHBOR TABLE

L	1	2	3	4	5	6	7	8	9	10	11	12
1	20	1	1	29	1	1	18	1	1	21	1	1
2	17	2	2	30	2	2	19	2	2	22	2	2
3	18	3	3	31	3	3	20	3	3	23	3	3
4	19	4	4	32	4	4	17	4	4	24	4	4
5	24	5	5	17	5	5	22	5	5	25	5	5
6	21	6	6	18	6	6	23	6	6	26	6	6
7	22	7	7	19	7	7	24	7	7	27	7	7
8	23	8	8	20	8	8	21	8	8	28	8	8
9	28	9	9	21	9	9	26	9	9	29	9	9
10	25	10	10	22	10	10	27	10	10	30	10	10
11	26	11	11	23	11	11	28	11	11	31	11	11
12	27	12	12	24	12	12	25	12	12	32	12	12
13	32	13	13	25	13	13	30	13	13	17	13	13
14	29	14	14	26	14	14	31	14	14	18	14	14
15	30	15	15	27	15	15	32	15	15	19	15	15
16	31	16	16	28	16	16	29	16	16	20	16	16
17	36	45	34	37	32	30	22	24	4	13	2	5
18	33	46	35	38	29	31	23	21	1	14	3	6
19	34	47	36	39	30	32	24	22	2	15	4	7
20	35	48	33	40	31	29	21	23	3	16	1	8
21	40	33	38	41	20	18	26	28	8	1	6	9
22	37	34	39	42	17	19	27	25	5	2	7	10
23	38	35	40	43	18	20	28	26	6	3	8	11
24	39	36	37	44	19	17	25	27	7	4	5	12
25	44	37	42	45	24	22	30	32	12	5	10	13
26	41	38	43	46	21	23	31	29	9	6	11	14
27	42	39	44	47	22	24	32	30	10	7	12	15
28	43	40	41	48	23	21	29	31	11	8	9	16
29	48	41	46	33	28	26	18	20	16	9	14	1
30	45	42	47	34	25	27	19	17	13	10	15	2
31	46	43	48	35	26	28	20	18	14	11	16	3
32	47	44	45	36	27	25	17	19	15	12	13	4
33	52	61	50	53	48	46	38	40	20	29	18	21
34	49	62	51	54	45	47	39	37	17	30	19	22
35	50	63	52	55	46	48	40	38	18	31	20	23
36	51	64	49	56	47	45	37	39	19	32	17	24
37	56	49	54	57	36	34	42	44	24	17	22	25
38	53	50	55	58	33	35	43	41	21	18	23	26
39	54	51	56	59	34	36	44	42	22	19	24	27
40	55	52	53	60	35	33	41	43	23	20	21	28
41	60	53	58	61	40	38	46	48	28	21	26	29
42	57	54	59	62	37	39	47	45	25	22	27	30
43	58	55	60	63	38	40	48	46	26	23	28	31
44	59	56	57	64	39	37	45	47	27	24	25	32
45	64	57	62	49	44	42	34	36	32	25	30	17
46	61	58	63	50	41	43	35	33	29	26	31	18
47	62	59	64	51	42	44	36	34	30	27	32	19
48	63	60	61	52	43	41	33	35	31	28	29	20
49	68	77	66	69	64	62	54	56	36	45	34	37
50	65	78	67	70	61	63	55	53	33	46	35	38
51	66	79	68	71	62	64	56	54	34	47	36	39
52	67	80	65	72	63	61	53	55	35	48	33	40
53	72	65	70	73	52	50	58	60	40	33	38	41
54	69	66	71	74	49	51	59	57	37	34	39	42
55	70	67	72	75	50	52	60	58	38	35	40	43
56	71	68	69	76	51	49	57	59	39	36	37	44
57	76	69	74	77	56	54	62	64	44	37	42	45
58	73	70	75	78	53	55	63	61	41	38	43	46
59	74	71	76	79	54	56	64	62	42	39	44	47
60	75	72	73	80	55	53	61	63	43	40	41	48
61	80	73	78	65	60	58	50	52	48	41	46	33
62	77	74	79	66	57	59	51	49	45	42	47	34
63	78	75	80	67	58	60	52	50	46	43	48	35
64	79	76	77	68	59	57	49	51	47	44	45	36

```
    THIS IS PLANE NUMBER      1
  2   1   2   1
  1   2   1   2
  2   1   2   1
  1   2   1   2
  THIS IS PLANE NUMBER     5
2   0   2   0
0   2   0   2
2   0   2   0
0   2   0   2
```

APPENDIX G

SAMPLE OUTPUT OF `DIFFUSE'

The following output is a representation of the output that `DIFFUSE' generates to use in the analysis of the diffusion prcess.

```
                            OUTPUTED VALUES


IMAX   IMAX1    JMAX   JMAX1    KMAX   KMAX1  PLNMAX    LMAX   LMAX1
   4       3       4       3       6       5      16      96      80


                         RANDOM GENERATOR SEEDS
             854265425      1816688514


                            TIME CONSTANTS


      START TIME  END TIME
               1       100
```

```
                    INPUTED CRYSTAL


LATICE ATOM - 2       DIFFUSING ATOM - 1            VACANCY - 0


     SITE ATOM TYPE
        1         2
        2         1
        3         2
        4         1
        5         1
        6         2
        7         1
        8         2
        9         2
       10         1
       11         2
       12         1
       13         1
       14         2
       15         1
       16         2
       17         0
       18         2
       19         0
       20         2
       21         2
       22         0
       23         2
       24         0
       25         0
       26         2
       27         0
       28         2
       29         2
       30         0
       31         2
       32         0
       33         2
       34         0
       35         2
       36         0
       37         0
       38         2
       39         0
       40         2
       41         2
       42         0
       43         2
       44         0
       45         0
       46         2
       47         0
       48         2
       49         0
       50         2
       51         0
       52         2
       53         2
       54         0
       55         2
       56         0
       57         0
       58         2
       59         0
       60         2
       61         2
       62         0
       63         2
       64         0
```

NEAREST NEIGHBOR TABLE

L	1	2	3	4	5	6	7	8	9	10	11	12
1	20	1	1	29	1	1	18	1	1	21	1	1
2	17	2	2	30	2	2	19	2	2	22	2	2
3	18	3	3	31	3	3	20	3	3	23	3	3
4	19	4	4	32	4	4	17	4	4	24	4	4
5	24	5	5	17	5	5	22	5	5	25	5	5
6	21	6	6	18	6	6	23	6	6	26	6	6
7	22	7	7	19	7	7	24	7	7	27	7	7
8	23	8	8	20	8	8	21	8	8	28	8	8
9	28	9	9	21	9	9	26	9	9	29	9	9
10	25	10	10	22	10	10	27	10	10	30	10	10
11	26	11	11	23	11	11	28	11	11	31	11	11
12	27	12	12	24	12	12	25	12	12	32	12	12
13	32	13	13	25	13	13	30	13	13	17	13	13
14	29	14	14	26	14	14	31	14	14	18	14	14
15	30	15	15	27	15	15	32	15	15	19	15	15
16	31	16	16	28	16	16	29	16	16	20	16	16
17	36	45	34	37	32	30	22	24	4	13	2	5
18	33	46	35	38	29	31	23	21	1	14	3	6
19	34	47	36	39	30	32	24	22	2	15	4	7
20	35	48	33	40	31	29	21	23	3	16	1	8
21	40	33	38	41	20	18	26	28	8	1	6	9
22	37	34	39	42	17	19	27	25	5	2	7	10
23	38	35	40	43	18	20	28	26	6	3	8	11
24	39	36	37	44	19	17	25	27	7	4	5	12
25	44	37	42	45	24	22	30	32	12	5	10	13
26	41	38	43	46	21	23	31	29	9	6	11	14
27	42	39	44	47	22	24	32	30	10	7	12	15
28	43	40	41	48	23	21	29	31	11	8	9	16
29	48	41	46	33	28	26	18	20	16	9	14	1
30	45	42	47	34	25	27	19	17	13	10	15	2
31	46	43	48	35	26	28	20	18	14	11	16	3
32	47	44	45	36	27	25	17	19	15	12	13	4
33	52	61	50	53	48	46	38	40	20	29	18	21
34	49	62	51	54	45	47	39	37	17	30	19	22
35	50	63	52	55	46	48	40	38	18	31	20	23
36	51	64	49	56	47	45	37	39	19	32	17	24
37	56	49	54	57	36	34	42	44	24	17	22	25
38	53	50	55	58	33	35	43	41	21	18	23	26
39	54	51	56	59	34	36	44	42	22	19	24	27
40	55	52	53	60	35	33	41	43	23	20	21	28
41	60	53	58	61	40	38	46	48	28	21	26	29
42	57	54	59	62	37	39	47	45	25	22	27	30
43	58	55	60	63	38	40	48	46	26	23	28	31
44	59	56	57	64	39	37	45	47	27	24	25	32
45	64	57	62	49	44	42	34	36	32	25	30	17
46	61	58	63	50	41	43	35	33	29	26	31	18
47	62	59	64	51	42	44	36	34	30	27	32	19
48	63	60	61	52	43	41	33	35	31	28	29	20
49	68	77	66	69	64	62	54	56	36	45	34	37
50	65	78	67	70	61	63	55	53	33	46	35	38
51	66	79	68	71	62	64	56	54	34	47	36	39
52	67	80	65	72	63	61	53	55	35	48	33	40
53	72	65	70	73	52	50	58	60	40	33	38	41
54	69	66	71	74	49	51	59	57	37	34	39	42
55	70	67	72	75	50	52	60	58	38	35	40	43
56	71	68	69	76	51	49	57	59	39	36	37	44
57	76	69	74	77	56	54	62	64	44	37	42	45
58	73	70	75	78	53	55	63	61	41	38	43	46
59	74	71	76	79	54	56	64	62	42	39	44	47
60	75	72	73	80	55	53	61	63	43	40	41	48
61	80	73	78	65	60	58	50	52	48	41	46	33
62	77	74	79	66	57	59	51	49	45	42	47	34
63	78	75	80	67	58	60	52	50	46	43	48	35
64	79	76	77	68	59	57	49	51	47	44	45	36

```
PARAMETERS FOR TIMESTEP   100


     PLANE   CON  T-  C/VOL T-      FLUX  1ST DERV  2ND DERV TIME DERV    CON T+  C/VOL T+
         1         8   1.00000   0.00000   0.00000   0.00000   0.00000         8   1.00000
         2         6   0.75000   0.00000  -0.12500   0.00000   0.00000         6   0.75000
         3         6   0.75000   0.12500  -0.31250  -0.12500  -0.12500         5   0.62500
         4         1   0.12500   0.00000  -0.37500   0.12500   0.12500         2   0.25000
         5         0   0.00000   0.00000  -0.06250   0.00000   0.00000         0   0.00000
```

```
AVERAGE PARAMETERS FOR TIMESTEP    1TO  100


  PLANE   AVG CONC  AVG C/VOL   AVG FLUX    1ST DER    2ND DER   TIME DER DIF COEF 1 DIF COEF 2 DIF CO DER TIME DER 7
      1    7.99999    1.00000    0.08500    0.00000    0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
      2    6.00997    0.75125    0.07750   -0.23812    0.00000    0.00750    0.00000    0.00000    0.00000    0.00000
      3    4.18997    0.52375    0.07125   -0.21625    0.03187    0.00625    0.32948    0.19608    0.39427   -0.07476
      4    2.54998    0.31875    0.06875   -0.17437    0.02844    0.00250    0.39427    0.08791    0.00000    0.00000
      5    1.39999    0.17500    0.06875   -0.15937    0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
```

LIST OF REFERENCES

Askeland, D.R., *The Science and Engineering of Materials*, pp. 95-103, Brooks/Cole Engineering Division, 1984.

Bakker, H., Stolwijk, N.A., van der Mey, L. and Zuurendonk, T.J., "Computer Simulation for Material Applications", *Nucl. Metall*, p.96, 1976.

Bennett, C.H. and Alder, B.J., "Persistence of Vacancy Motion in Hard Sphere Crystals", *J. Phys. Chem. Solids*, Vol 32, pp. 2111-2122, 1971.

Bowker, M. and King, D.A., *Surface Science*, Vol 71, p. 7[illegible]0, 1978.

Collins, R. (Professor, University of Salford) to Harrison, D.E. (Professor, Naval Postgraduate School), *Fundamental Studies in Physics*, Private Communication, 27 September 1985.

Crank, J., *The Mathematics of Diffusion*, p. 2, Oxford University Press, 1975.

Darken, L.S., "Diffusion, Mobility and Their Interrelation through Free Energy in Binary Metallic Systems", *Transactions*, American Institute of Mining and Metallurgical Engineers, Vol. 175, p. 184, 1948.

de Bruin, H.J. and Murch, G.E., *Philosophical Magazine A*, Vol 27, p. 1475, 1973.

Einstein, A., "Uber die von der molekular-kinetischen Theorie der Warme-geforderte Bewegung von in ruhenden Flussigkeiten suspendieren Teilchen", *Annalen der Physik*, Vol. 17, p. 549, 1905.

Fick, A., "Uber Diffusion", *Pogg. Ann.*, Vol. 94, p. 59, 1855.

Flinn, P.A. and McManus, G.M., *Physical Review*, Vol. 124, p. 54, 1961.

Ghez, R. and Langlois, W.E., "More on the Concentration Dependence of Fick's Laws", *American Journal of Physics*, To be published July 1986.

Girifalco, L.A., Atomic Migration in Crystals, Blaisdell Publishing Company, 1964.

Gordon, A.J. and Ford, R.A., The Chemist's Companion, pp. 114-115, John Wiley & Sons, Inc., 1972.

Guy, A.G., Transactions, Japanese Institute of Metalurgy, Vol 19, p. 483, 1978.

Guy, A.G., Cooper, W.D. and Poole R.L., "Material Science, Vol 3, p. 103, 1977.

Harrison, D.E., Jr., Simulation of Physical Systems, Unpublished notes for PH3911 at the Naval Postgraduate School, 1985.

Hartley, G.S., "Diffusion and Swelling of High Polymers, I", Transactions, Faraday Society, Vol. 42, p. 6, 1946.

King, G.W., Ind. Eng. Cem., Vol. 43, p. 2475, 1951.

Kirkendall, E.O., "Diffusion of Zinc in Alpha Brass", Transactions, American Institute of Mining and Metallurgical Engineers, Vol. 147, p. 104, 1942.

Le Claire, A.D., Physical Chemistry: An Advanced Treatise, Vol 10, Academic Press, p. 261, 1970.

Manning, J.R., Diffusion Kinetics for Atoms in Crystals, pp. 2-10, D. Van Nostrand Company, Inc., 1968.

Matano, C., "On the Relation Between the Diffusion Coefficients and Concentrations of Solid Metals", Journal of Physics (Japan), Vol 8, p. 109, 1933.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E., J. Chem. Phys., Vol. 21, p. 1087, 1953.

Murch, G.E., "Chemical Diffusion in Highly Defective Solids", Philosophical Magazine A, Vol 41, pp. 157-163, 1980.

Murch, G.E., Fast Ionic Transport in Solids, 1981.

Murch, G.E. and Nowick, A.S., "Simulation in Crystalline Solids", Diffusion in Crystalline Solids, pp. 379-427, 1984.

Murch, G.E. and Thorn R.J., "Calculation of the Diffusion Correlation Factor by Monte Carlo Methods", _Philosophical Magazine A_, Vol 39, pp. 673-677, 1979.

Slavin, A.J. and Underhill, P.R., _American Journal of Physics_ Vol 52, p. 376, 1984.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5000	2
3.	Department Chairman, Code 61 Department of Physics Naval Postgraduate School Monterey, California 93943-5000	2
4.	Dr. D.E. Harrison, Jr., Code 61Hx Department of Physics Naval Postgraduate School Monterey, California 93943-5000	4
5.	Dr. Lester Ingber, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93943-5000	3
6.	Lt. M.R. Polnaszek, USN 349 Condon Dr. Charleston, South Carolina 29412	4

279
18070 2

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943-5002

219316

Thesis
P732
c.1

Polnaszek
Aspects of simulating interstitial diffusion in a face-centered-cubic lattice.

219316

Thesis
P732
c.1

Polnaszek
Aspects of simulating interstitial diffusion in a face-centered-cubic lattice.

Aspects of simulating interstitial diffu
3 2768 000 67350 3
DUDLEY KNOX LIBRARY